بسم الله الرحمن الرحيم

بعد تقديمى ل (مذكرة فى الحاسب الآلى للصف الأول الثانوى ترم ثانى) فى ١٦ / ٤ / ٢٠٠٩ يسرنى أن أتقدم بهذا الاجتهاد المتواضع لأبنائي تلاميذ الصف الثالث الإعدادي بجمهورية مصر العربية و لكل طلاب العلم فى كل مكان . و لقد حاولت تقديم مواضيع منهج الكمبيوتر بشكل مبسط ليحقق أكبر قدر من الاستفادة فى فهم المنهج.



ختاما أدعو الله أن ينال هذا المجهود رضاكم و أتمنى دعائكم...

لأى استفسار يرجى مراسلتى على البريد الالكترونى

المؤلف

أسامة مصطفى نـصر

مدرسة الثانـوية الجديدة للبنـات بشبين الكوم

ت :– 01001354240

البريد الالكترونى :– osamanasr@yahoo.com

الفصل الأول

١ـ نظام المعلومات

هو أى نظام يجمع بين الكمبيوتر و الأفراد و يسمح لأى مؤسسة بجمع و تخزين البيانات ثم معالجتها للحصول على المعلومات.

و أى ***نظام معلومات*** يتكون على ثلاثة عناصر هى ١ـ المدخلات (البيانات) ٢ـ المعالجة

٣ ـ المخرجات (المعلومات) و لفهم ذلك نفترض مثال جمع عددين و الذى يمكن تمثيله كالآتى:-

المدخلات (البيانات)	المعالجة (عملية الجمع)	المخرجات (المعلومات)
25 , 7	25 + 7	32

- البيانات :- هى مجموعة الحقائق المجردة- المادة الخام أو المعطيات الأولية - التى يتم تجميعها و تخزينها بواسطة نظام المعلومات و عن طريق معالجتها نحصل على المعلومات . و تأخذ البيانات أشكالا متعددة منها الأرقام ، الحروف ، الصور ، و الأصوات و الفيديو. و هى فى حد ذاتها لا تعطى معنى مفيد ذو دلالة قاطعة.

و من أمثلة البيانات : الأسماء ، درجات الحرارة ، الأسعار ، درجة الاختبار ، وظيفة أو مهنة ، نوع وسيلة مواصلات (قطار – سيارة – باخرة) – لون الشعر - صوتك

- المعلومات :- هى ناتج معالجة البيانات . و معالجة البيانات يقصد بها إجراء أى عملية حسابية (جمع – طرح – ضرب – قسمة ... الخ) أو منطقية (مقارنة – اكبر من – اصغر من .... الخ) أو ترتيب أو تجميع البيانات لتعطى معنى مفيد.

و من أمثلة ذلك إذا عرفنا أن طالب معين حصل على ٣٥ درجة فى اختبار اللغة العربية فلن يكون لذلك معنى إلا إذا عرفنا النهاية العظمى و الصغرى لدرجات اختبار اللغة العربية لتقرير ما إذا كان ناجحا أم راسبا و لتقرير التقدير الذى حصل عليه (ضعيف – مقبول – جيد – جيد جدا - ممتاز) يلزم معرفة حدود كل تقدير .

من هذا المثال نستنتج أنه لاستخلاص المعلومة يجب إجراء عملية ما و هى هنا المقارنة على البيانات المتاحة للحصول على المعلومة.

و عندما تكون البيانات كثيرة تكون العمليات المطلوبة كثيرة مما يستدعى إجراء ملايين العمليات الحسابية و حتى نضمن سرعة و دقة تنفيذ العمليات فإننا نعتمد على الكمبيوتر. و لتنفيذ ما هو مطلوب منه يحتاج الكمبيوتر أن نزوده بمجموعة الأوامر اللازمة لتنفيذ هذه العمليات و هذه المجموعات من الأوامر تسمى برامج و عملية إنشاء هذه البرامج تسمى عملية برمجة.

١ـ العمليات التى يقوم بها الكمبيوتر

يمكن تصنيف العمليات التى يقوم بها الكمبيوتر إلى ست أنواع من العمليات هى:-

١ـ إدخال أو قراءة البيانات (من أى وحدة من وحدات الإدخال)

٢ـ تخزين البيانات فى الذاكرة الداخلية (RAM)

٣- إجراء عمليات حسابية على البيانات (جمع – طرح – ضرب – قسمة)

٤- مقارنة بين قيمتين و الاختيار بين بديلين لتنفيذ أحدهما .(أكبر من – أصغر من - يساوى – أكبر من أو يساوى.... الخ)

٥- تكرار تنفيذ أمر معين أو مجموعة من الأوامر لأى عدد من المرات.

٦- إخراج النتائج (على أى وحدة من وحدات الإخراج).

و فيما يلى شرح لهذه العمليات:

١- ***إدخال البيانات (قراءة البيانات) :-*** لكى يتمكن الكمبيوتر من معالجة البيانات لتحويلها إلى معلومات يجب أولا أن تكون متاحة له و هذا يتم عن طريق وحدات الإدخال المختلفة حسب نوع البيان (لوحة المفاتيح – الفأرة – الماسح الضوئى .. الخ) و كذلك عن طريق وسائط التخزين مثل القرص الصلب Hard Disk و القرص المرن Floppy Disk و القرص المدمج CD.[وسائط التخزين تعتبر أيضا من وحدات الإدخال و الإخراج معا فعند القراءة منها تعتبر وحدة إدخال و عند الكتابة عليها تعتبر وحدة إخراج]

٢- *تخزين البيانات فى ذاكرة الكمبيوتر الداخلية :-* يتم تخزين البيانات التى تم إدخالها للكمبيوتر فى الذاكرة المؤقتة- الداخلية- (RAM) التى ترقم جميع خلاياها و يعطى كل مكان فى الذاكرة يحمل بيانا اسما محددا و يتم التعامل مع البيان من خلال الاسم و يمكن أن يتم تغيير محتويات الاسم و هى فى هذه الحالة البيان و هذا ما يسمى بالمتغيرات Variables و كمثال على ذلك إذا تم حجز مكان بالذاكرة تحت الاسم A و تم وضع الرقم 7 داخل هذا المتغير و متغير آخر باسم B و تم وضع العدد ٢٥ به و تم جمع العددين و تخزين النتيجة فى المتغير C و قيمتها ٣٢ فيمكن تمثيل ذلك بالشكل التالى:

	4		3		2		١
A	8		7		6		5
7							
	12		11	B	10		9
				25			
	16	C	15		14		13
		32					

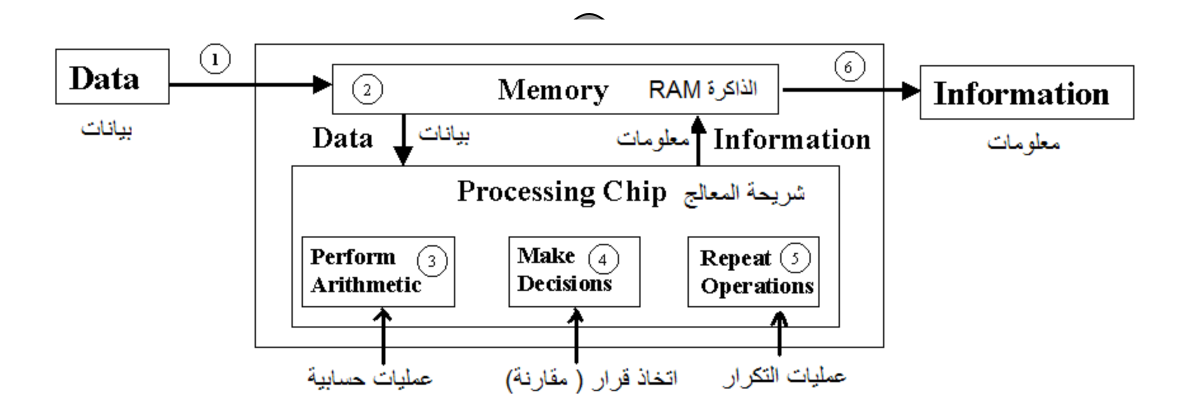
إجراء العمليات الحسابية على البيانات:- يستطيع الكمبيوتر بواسطة المعالج الدقيق (Microprocessor) أن يقوم بتنفيذ جميع العمليات الحسابية على البيانات (جمع – طرح – ضرب..... الخ) حسب الأوامر

٣- ***المقارنة بين قيمتين :*** يستطيع الكمبيوتر المقارنة بين قيمتين أو :ثر و لهذا يمكن عن طريق هذه المقارنة اختيار أمر لتنفيذه و تجاهل آخر أو العكس مثال ذلك لتحديد ما إذا كان العدد موجبا أو سالبا فإننا نقارن قيمة هذا العدد بالصفر فإذا كان أكبر من الصفر فهو موجب و إذا كان أصغر من الصفر فهو سالب.

٤- ***تكرار تنفيذ أمر أو مجموعة أوامر لعدد معين من المرات:-*** يستطيع الكمبيوتر أن يقوم بتنفيذ أمر واحد أو مجموعة أوامر لعدد صغير أو كبير من المرات و يمكن نتيجة لخطأ ما أن يستمر التنفيذ إلى ما لا نهاية أو حتى إيقاف أو إعادة تشغيل الكمبيوتر.

٥- ***إخراج النتائج:*** بعد معالجة البيانات يصبح الناتج هو المعلومات و التى تكون جاهزة للإخراج سواء فى شكل عرض على الشاشة أو طباعة على الورق أو تخزينها على وسائط تخزين.

و يمكن تمثيل العمليات الست فى هذا الشكل:-



- **البرامج و البرمجة**

البرمجة هى طريقة لحل المسائل

Algorithm (الخوارزمية) :- **خطوات مرتبة ترتيبا منطقيا يؤدى إلى حل المسألة.** و فى الحقيقة فإن كل الخطوات التى نقوم بها لانجاز عمل ما بنجاح فى حياتنا اليومية هو Algorithm. و **لابد من كتابة Algorithm لكل برنامج قبل كتابة البرنامج حتى لا نتعرض للخطأ.**

- **لغات البرمجة Programming Languages**

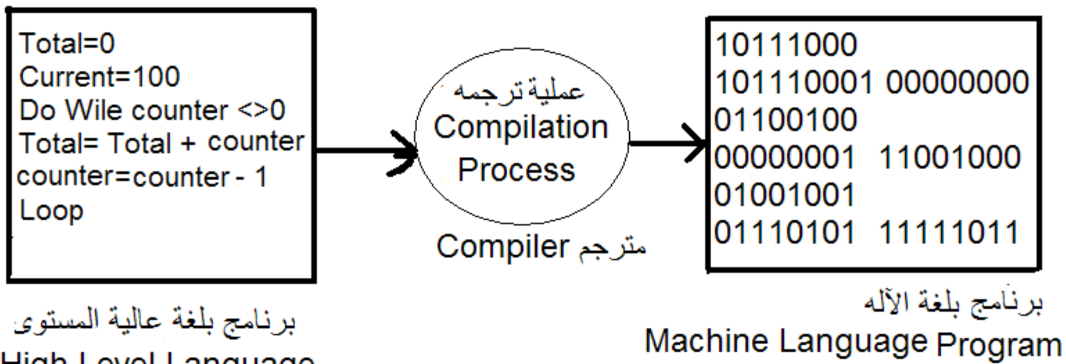
الكمبيوتر لا يفهم أى لغة من اللغات التى يتحدث بها البشر سواء كانت الانجليزية أو العربية أو غيرها ، إنما يفهم لغة واحدة تسمى لغة الآلة و مفرداتها هى (0,1) و الصفر يعنى أن الخلية Bit قيمتها صفر(الدائرة لا تحتوى شحنة كهربية) و 1 تعنى أن الخلية Bit قيمتها 1 (الدائرة تحوى شحنة كهربية). و لتسهيل التعامل مع الكمبيوتر و برمجته تم اختراع لغات البرمجة عالية المستوى و هى لغات برمجة تكتب بحروف انجليزية و تتبع قواعد محددة فى كتابتها لابد من إتباعها عند كتابة البرنامج- و هو أمر مشابه للكتابة أو التحدث بأى لغة يستعملها البشر فيما بينهم- و يقوم برنامج آخر وسيط بترجمة أو تفسير هذه الأوامر و يسمى مترجم أو مفسر حسب نوع لغة البرمجة إلى لغة الآلة ليقوم الكمبيوتر بتنفيذها.

و كما تتعدد لغات البشر تتعدد لغات برمجة الكمبيوتر و من أمثلة لغات البرمجة عالية المستوى الكوبول COBOL الفورتران Fortran الجافا JAVA و الـ ++C و البيزك Basic و تتعدد إصدارات لغة البيزك فمنها الأقدم مثل Gw

basic , Quick Basic و التى كانت تعمل فى بيئة نظام التشغيل Dos و الأحدث منه Visual Basic و الذى يعمل فى بيئة نظام التشغيل ويندوز .

و تتميز لغة Visual Basic .Net باستخدام رموز و كلمات يسهل فهمها و تذكرها لتسهيل عملية البرمجة. كما يتيح هذا الإصدار للمبرمج الاستعانة ببعض الأدوات و الأشكال الجاهزة لعمل واجهة جيدة تسهل عمل مستخدم البرنامج. كما تنفرد بأنها تمتلك مفسر Interpreter و مترجم Compiler معا .

و الشكل التالى يمثل عملية الترجمة التى يمر بها البرنامج



- **ما الفرق بين المفسر و المترجم ؟**

- **أولا المفسر(Interpreter) :-** يقوم المفسر بتحويل الأوامر من لغة البرمجة عالية المستوى الى لغة الآلة حتى يتم تنفيذها . و يقوم بمراجعة خطوات البرنامج بداية من أول خطوة فإذا وجد الخطوة صحيحة لغويا(حسب قواعد لغة البرمجة) يقوم بتحويلها الى لغة الآلة(تنفيذها) و ينتقل للخطوة التالية و هكذا حتى يتم الانتهاء من تنفيذ كامل خطوات البرنامج. و يعيب المفسر البط ء فى التنفيذ بينما يتميز بسهولة كشف الأخطاء.
- **ثانيا المترجم (Compiler)** يقوم بمراجعة كافة خطوات البرنامج دفعة واحدة و لا يقوم بتنفيذ البرنامج (تحويله إلى لغة الآلة) إلا بعد تصحيح كافة الأخطاء. و يتميز المترجم بالسرعة فى التنفيذ و صعوبة تصحيح الأخطاء.

و لغة البرمجة VB.Net تستخدم المفسر أثناء عملية كتابة البرنامج(لسهولة كشف و تصحيح الأخطاء) بينما يستخدم المترجم لتنفيذ البرنامج(للاستفادة من سرعته).

و فيما يلى مقارنة بين كلا من المفسر و المترجم:-

Interpreter المفسر	Compiler المترجم
١- يقوم بتحويل تعليمات لغة البرمجة عالية المستوى الى لغة الآلة	١- يقوم بتحويل تعليمات لغة البرمجة عالية المستوى الى لغة الآلة
٢- يفسر أوامر البرنامج سطرا سطرا و فى حالة عدم	٢- يتم ترجمة البرنامج بالكامل للغة الآلة و لا ينفذ البرنامج

وجود خطأ يقوم بتنفيذه	إلا فى حالة خلوه من جميع الأخطاء
٣- يتوقف عن تنفيذ البرنامج عند السطر الموجود به خطأ	٣- يتوقف عن التنفيذ حتى يتم تصحيح جميع الأخطاء الموجودة بالبرنامج
٤- يظهر رسالة عند مكان الخطأ	٤- يعطى تقريرا إجماليا بالأخطاء
٥- يبدأ العمل فى تنفيذ البرنامج بعد إصلاح الخطأ و يستمر التنفيذ حتى ظهور أخطاء أخرى	٥- لا يبدأ العمل فى تنفيذ البرنامج حتى يتم تصحيح جميع الأخطاء.
٦- أبطأ فى التنفيذ	٦- أسرع فى التنفيذ
٧- أسهل فى اكتشاف الأخطاء	٧- أصعب فى اكتشاف الأخطاء

حل أسئلة الفصل الأول

س١- ما المقصود بنظام المعلومات.

ج١- هو أى نظام يجمع بين التكنولوجيا ممثلة في الكمبيوتر و الأفراد و يسمح لأى مؤسسة بجمع و تخزين البيانات ثم استخلاص معلومات من هذه البيانات.

س٢- ما الفرق بين البيانات و المعلومات؟

ج٢- المعلومات هى ناتج معالجة البيانات. حيث أن البيانات فى حد ذاتها لا تعطى معلومات هامة و لكن بعد إجراء بعض العمليات عليها تتحول إلى معلومات مفيدة.

س٣- كيف يمكنك تحويل البيانات إلى معلومات؟

ج٣- يمكن تحويل البيانات إلى معلومات بإجراء عملية معالجة لها (عمليات حسابية و منطقية)

س٤- أذكر الفرق بين اللغات عالية المستوى و لغة الآلة.

ج٤- اللغات عالية المستوى تستخدم تعليمات برمجية بلغة قريبة من لغة التحدث الطبيعية(اللغة الانجليزية) و سهلة التعلم و تستنفذ وقتا أقل من لغة الآلة فى كتابة البرنامج (عدد التعليمات أقل). بينما لغة الآلة تستخدم فقط الصفر و الواحد 1, 0 و هى صعبة التعلم و تستنفذ وقتا طويلا فى كتابة البرنامج (عدد التعليمات أكبر).

س٥- ما الفرق بين المفسر و المترجم؟

ج٥- المفسر يقوم بفحص خطوات البرنامج خطوة خطوة و عندما يتحقق من صحة الخطوة فإنه ينفذها و إذا وجد خطوة بها خطأ فإنه يتوقف عندها حتى يتم تصحيح الخطأ و عندها ينتقل للخطوة التالية و هكذا. أما المترجم فإنه يقوم بفحص جميع خطوات البرنامج دفعة واحدة وإذا وجد خطأ أو أخطاء فإنه لا يقوم بتنفيذ البرنامج حتى يتم تصحيح الخطأ أو الأخطاء جميعها.و عندما يكون البرنامج خالى من الأخطاء فإنه يبدأ تنفيذه.

الفصل الثانى و الفصل الثالث

مقدمة

يوجد عدة طرق لتمثيل طريقة حل مسألة (ترتيب خطوات الحل) الخوارزمية و منها أسلوبين أحدهما بصرى(بيانى) و هو خرائط التدفق و الآخر لفظى و هو السودو كود.

أولا:- خطوات حل مسألة :-

١- تحديد المسألة : أى تعريف المسألة و معرفة ما هو مطلوب على وجه الدقة.
٢- تعريف المسألة : و ذلك بإعطاء اسم لكل بيان يتم إدخاله (المعطيات - البيانات) و هذه تسمى بالمتغيرات Variables
٣- إهمال البيانات التى لن تؤثر فى الحل.
٤- تحديد المعلومات المطلوب الحصول عليها و تسمية المتغيرات الخاصة بالحل و وضع العلاقة بين المتغيرات (البيانات و المعلومات – المعطيات و النتيجة)
٥- يتم ترتيب خطوات الحل حسب أولوية التنفيذ اى تحويلها إلى Algorithm أو خوارزمية .

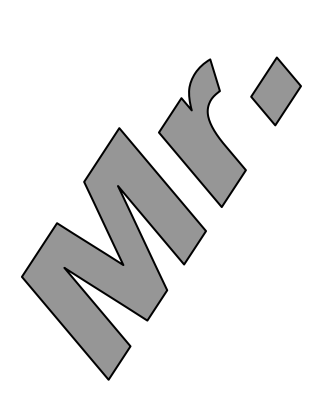
الخوارزمية الواردة فى الأمثلة الثلاثة التالية للدراسة و الفهم فقط و غير مقررة

مثال ١:- ما هو الأجر الصافى لعامل مرتبه الاجمالى الشهرى ٥٠٠ جنيه علما بأنه يتم خصم ١٠% من قيمة المرتب للضرائب و ٢% للتأمين الصحى.

الحل:-

١- تحديد المسألة: صافى المرتب الذى سيحصل عليه العامل
٢- تعريف المسألة (المعطيات – البيانات): اجمالى المرتب Total خصم الضرائب Tax خصم التأمين الصحى Health .
٣- ---------(لا يوجد بيانات مهملة)
٤- المعلومات المطلوب الحصول عليها هى صافى المرتب NetWage و = اجمالى المرتب – خصم الضرائب – خصم التأمين الصحى. أى

NetWage = Total – Tax – Health

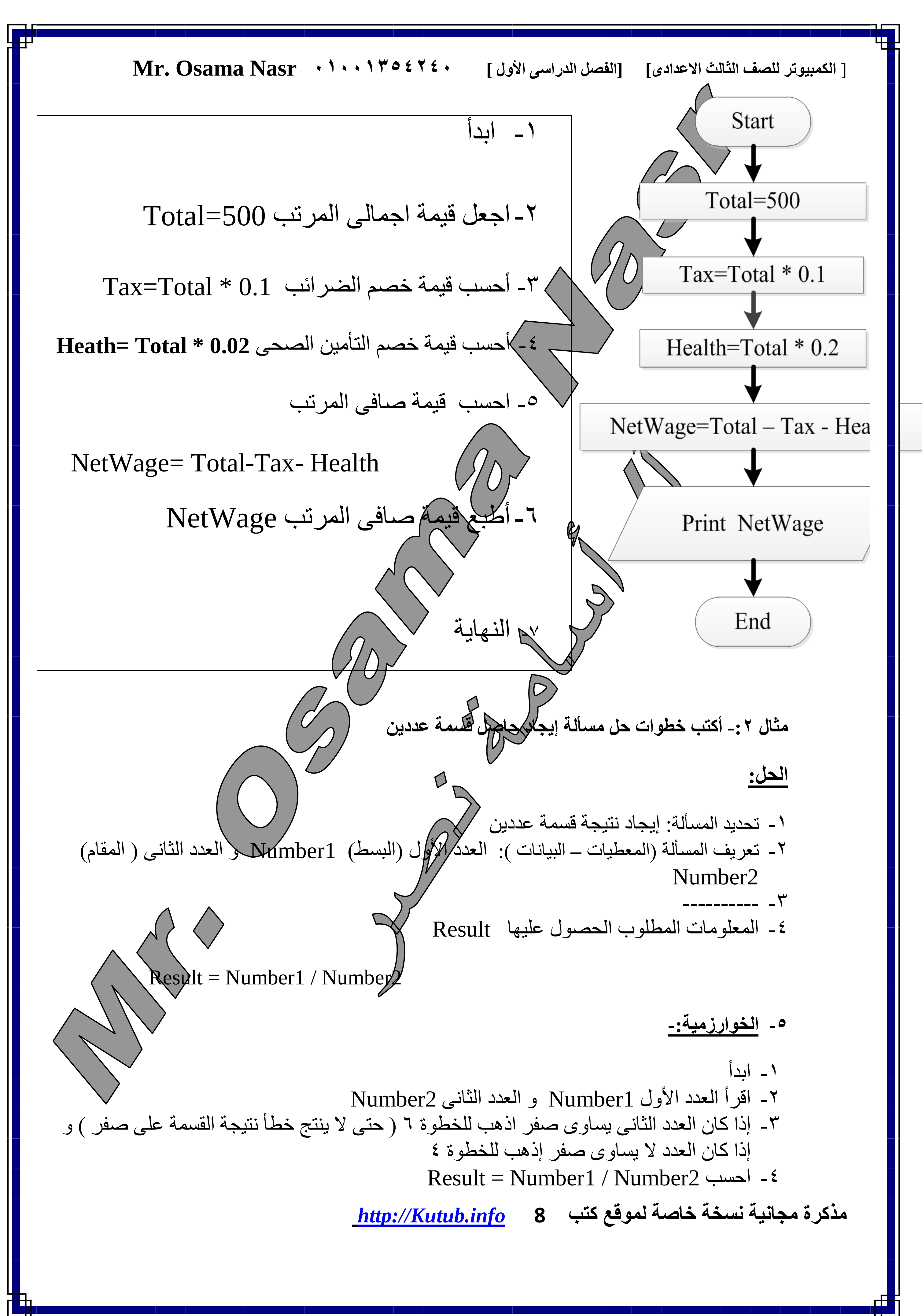


٥- **الخوارزمية :**

١- ابدأ
٢- اجعل قيمة اجمالى المرتب Total=500
٣- أحسب قيمة خصم الضرائب Tax=Total * 0.1
٤- أحسب قيمة خصم التأمين الصحى Heath= Total * 0.02
٥- احسب قيمة صافى المرتب

NetWage= Total-Tax-Health

٦- أطبع قيمة صافى المرتب NetWage
٧- النهاية



١- ابدأ

٢- اجعل قيمة اجمالى المرتب Total=500

٣- أحسب قيمة خصم الضرائب Tax=Total * 0.1

٤- أحسب قيمة خصم التأمين الصحى **Heath= Total * 0.02**

٥- احسب قيمة صافى المرتب

NetWage= Total-Tax- Health

٦- أطبع قيمة صافى المرتب NetWage

٧- النهاية

مثال ٢ :- أكتب خطوات حل مسألة إيجاد حاصل قسمة عددين

<u>الحل:</u>

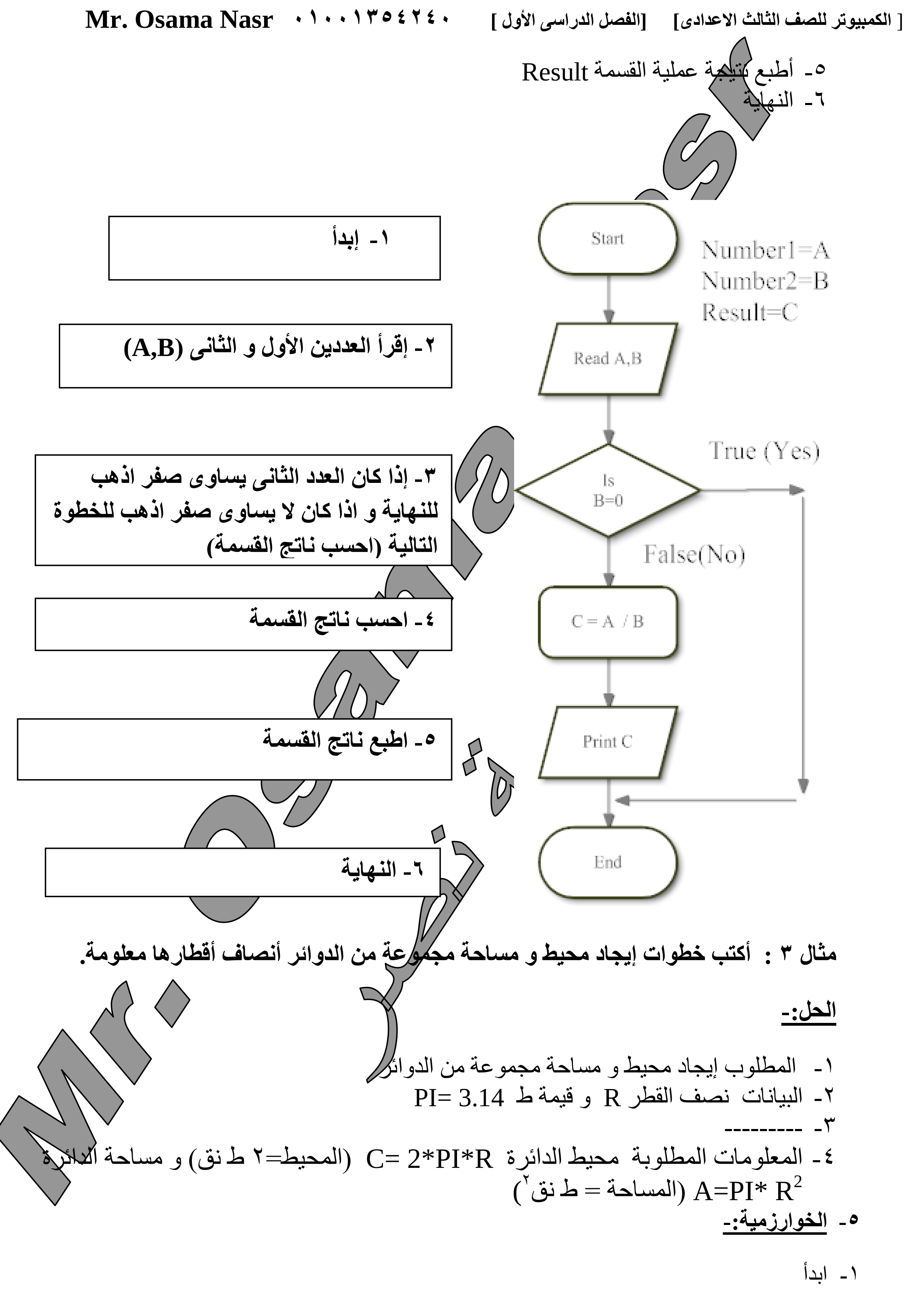
١- تحديد المسألة: إيجاد نتيجة قسمة عددين
٢- تعريف المسألة (المعطيات – البيانات): العدد الأول (البسط) Number1 و العدد الثانى (المقام) Number2
٣- ----------
٤- المعلومات المطلوب الحصول عليها Result

Result = Number1 / Number2

٥- **<u>الخوارزمية:-</u>**

١- ابدا
٢- اقرأ العدد الأول Number1 و العدد الثانى Number2
٣- إذا كان العدد الثانى يساوى صفر اذهب للخطوة ٦ (حتى لا ينتج خطأ نتيجة القسمة على صفر) و إذا كان العدد لا يساوى صفر إذهب للخطوة ٤
٤- احسب Result = Number1 / Number2

٥- أطبع نتيجة عملية القسمة Result
٦- النهاية



مثال ٣ : أكتب خطوات إيجاد محيط و مساحة مجموعة من الدوائر أنصاف أقطارها معلومة.

الحل:-

١- المطلوب إيجاد محيط و مساحة مجموعة من الدوائر
٢- البيانات نصف القطر R و قيمة ط PI= 3.14
٣- ---------
٤- المعلومات المطلوبة محيط الدائرة C= 2*PI*R (المحيط=٢ ط نق) و مساحة الدائرة A=PI* R^2 (المساحة = ط نق٢)

٥- **الخوارزمية:-**

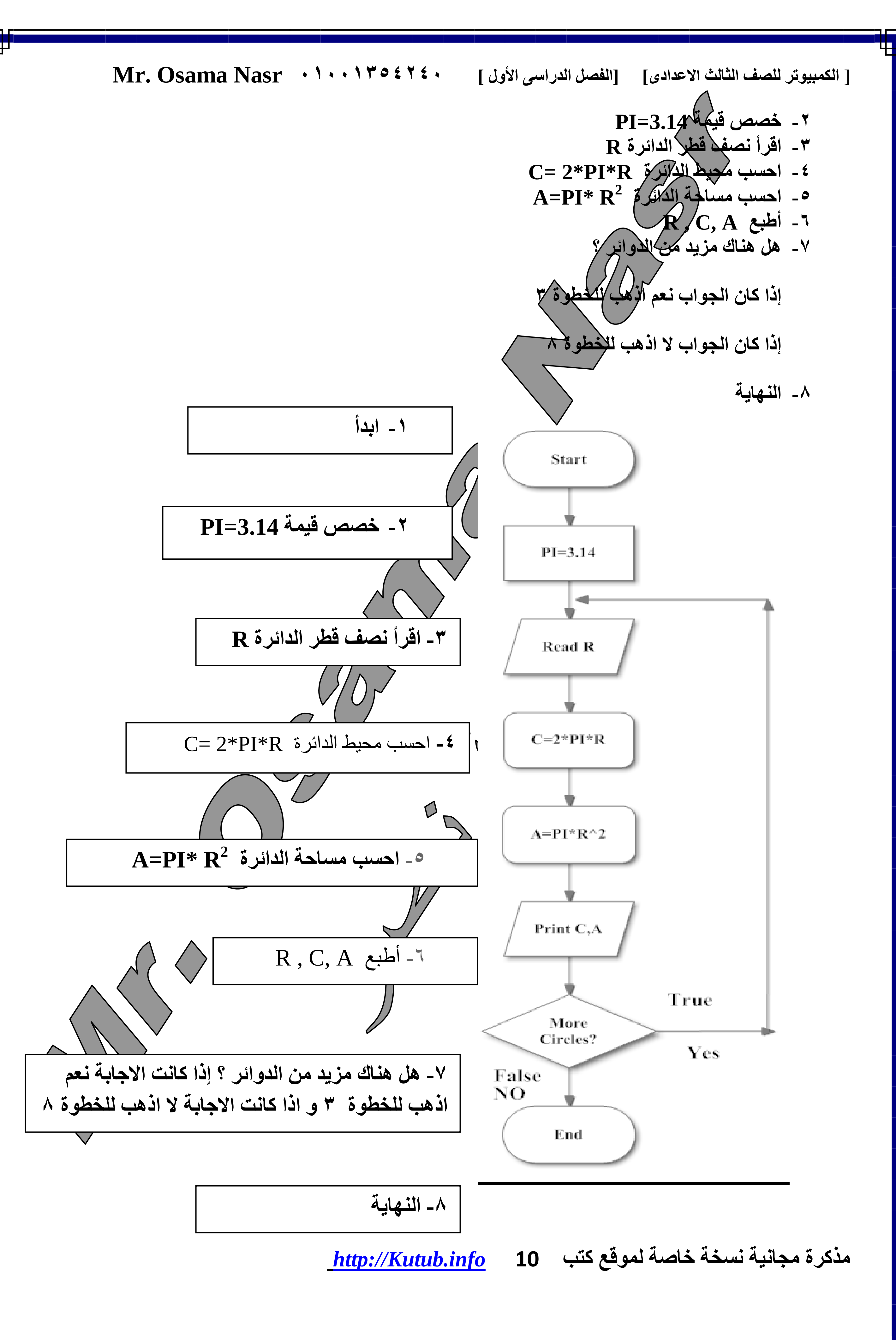
١- ابدأ

٢- خصص قيمة PI=3.14
٣- اقرأ نصف قطر الدائرة R
٤- احسب محيط الدائرة C= 2*PI*R
٥- احسب مساحة الدائرة $A=PI*R^2$
٦- أطبع R , C, A
٧- هل هناك مزيد من الدوائر ؟

إذا كان الجواب نعم اذهب للخطوة ٣

إذا كان الجواب لا اذهب للخطوة ٨

٨- النهاية



١- ابدأ

٢- خصص قيمة PI=3.14

٣- اقرأ نصف قطر الدائرة R

٤- احسب محيط الدائرة C= 2*PI*R

٥- احسب مساحة الدائرة $A=PI*R^2$

٦- أطبع R , C, A

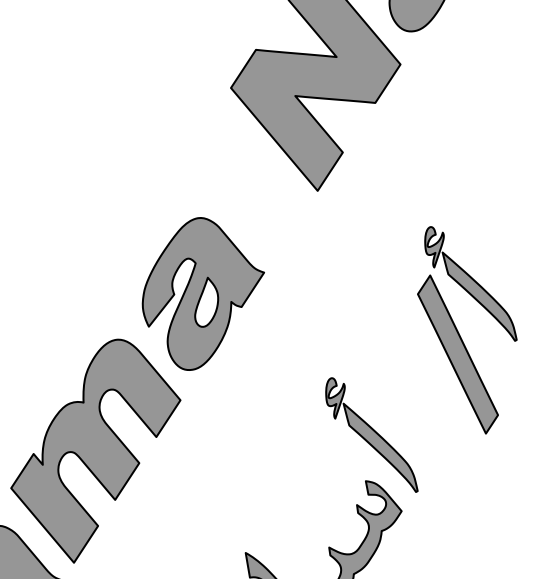
٧- هل هناك مزيد من الدوائر ؟ إذا كانت الاجابة نعم اذهب للخطوة ٣ و اذا كانت الاجابة لا اذهب للخطوة ٨

٨- النهاية

خرائط التدفق

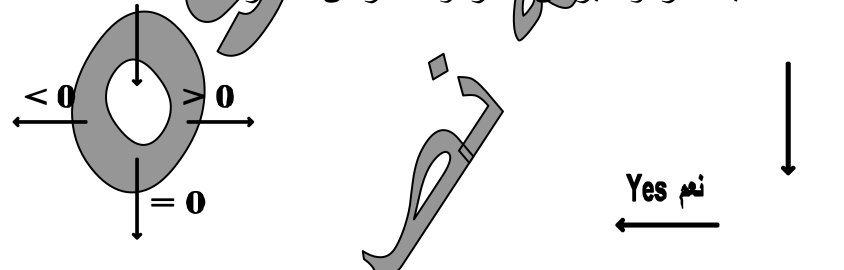
تعريف: خرائط التدفق هي تمثيل بياني يعتمد علي الرسم لتوضيح ترتيب العمليات اللازمة لحل مسألة.

الأشكال الخمسة الأساسية الأكثر استخداما في خرائط التدفق و الغرض المستخدمة من أجله



خط الاتجاه و هو خط مستقيم به سهم يحدد اتجاه سير العمليات

مع ملاحظة أن رمز اتخاذ القرار ينتج عنه إما نتيجتين فى حالة إذا كان المطلوب اتخاذ قرار بنعم أو لا أو ثلاث نتائج مثل مقارنة عدد ما بالصفر أو أكبر من صفر أو أصغر من الصفر



مميزات خرائط التدفق:

١. **الاتصال :** حيث أن الأشكال المستخدمة بها نمطية فإنها وسيلة سهلة لشرح خطوات الحل للآخرين.
٢. **التحليل الفعال :** يمكن لخرائط التدفق أن تمثل حل المسألة بطريقة فعالة (أكثر سهولة)
٣. **توثيق صحيح :** تعتبر خرائط التدفق توثيق جيد للبرامج.
٤. **تكويد كفء:** تساعد خرائط التدفق بعد رسمها فى كتابة البرامج بطريقة فعالة
٥. **تصحيح الأخطاء:** تساعد خرائط التدفق فى تتبع خطوات الحل لاكتشاف الأخطاء.
٦. **كفاءة إصلاح البرنامج :** إصلاح البرنامج يصبح سهلا بمساعدة خرائط التدفق.

عيوب خرائط التدفق:

١- **أسلوب معقد:** إذا كانت المسألة معقدة فإن رسمها غالبا ما يكون معقدا.
٢- **صعوبة إجراء تعديلات:** حيث ان إجراء تعديل ما ربما يحتاج إلى إعادة رسم الخريطة.
٣- **النسخ:** إذا كان لا يمكننا رسم رموز الخرائط فإن نسخ الخرائط يصبح مشكلة.
٤- **كثرة التفاصيل :** عند تتبع حل مسألة باستخدام خرائط التدفق فيمكن لكثرة التفاصيل أن تبعدنا عن الحل.

السودو كود Pseudocode

تعريف:- السودو كود هى طريقة لعرض خطوات حل مسألة معينة باستخدام أوامر مختصرة.

كلمة سودو كود تعنى الكود الزائف أو الكود الغير حقيقى (Pseudo + Code) و الكود يعنى البرنامج المكتوب بإحدى لغات البرمجة.و لهذا يصبح المفهوم الأوضح ***هو البرنامج الغير حقيقى***.و ذلك لأنه لم يكتب فعليا بلغة برمجة إنما كتب بلغة قريبة منها و يكتب باللغة الانجليزية لأن جميع لغات البرمجة تقريبا تستخدم اللغة الانجليزية .

مميزات السودو كود:

يتميز السودو كود بأنه لا يشتمل على أشكال نمطية يجب حفظها كم فى خرائط التدفق و لكنه مبنى على مجموعة من القواعد القليلة و سهلة الاستيعاب و التطبيق.

قواعد كتابة السودو كود :-

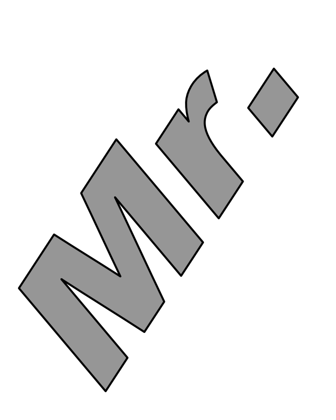
١- **اختيار أسماء للمتغيرات ذات معنى بحيث يعبر الاسم عن المحتويات و دورة فى الحل.**
٢- **كتابة الأوامر بطريقة واضحة لا غموض فيها.**

ملاحظة:- يجب ملاحظة القاعدة الثانية و هى الأهم فى كتابة السودو كود لأنه لا توجد الفاظ محددة أو جمل محددة انما العبرة بوضوح السودو كود و تأديته للغرض الذى كتب من أجله.

مزايا و عيوب السودو كود:-

١- سهل الفهم حيث أنه يستخدم الكلمات الانجليزية العادية
٢- لا يستعمل رموزا خاصة.
٣- ليس له قواعد خاصة.
٤- من السهل تحويله إلى برنامج باستعمال احدى لغات البرمجة.

أما أهم عيوبه فإنه قد يكون طويلا فى حالة المسائل المعقدة.



الأنماط المختلفة للبرامج و طرق تمثيلها

أولا :- النمط التتابعى (التسلسلى):-

و فى هذا النمط تكون الخطوات متتابعة و الشكل العام لتمثيلها بخرائط التدفق و السودو كود يأخذ الشكل التالى:-

التمثيل بالسودو كود		التمثيل بخرائط التدفق
Statement 1	الخطوة رقم ١	خطوة رقم ١
Statement 2	الخطوة رقم ٢	خطوة رقم ٢
Statement 3	الخطوة رقم ٣	خطوة رقم ٣
....	....	
....	....	
Statement n		خطوة رقم ن

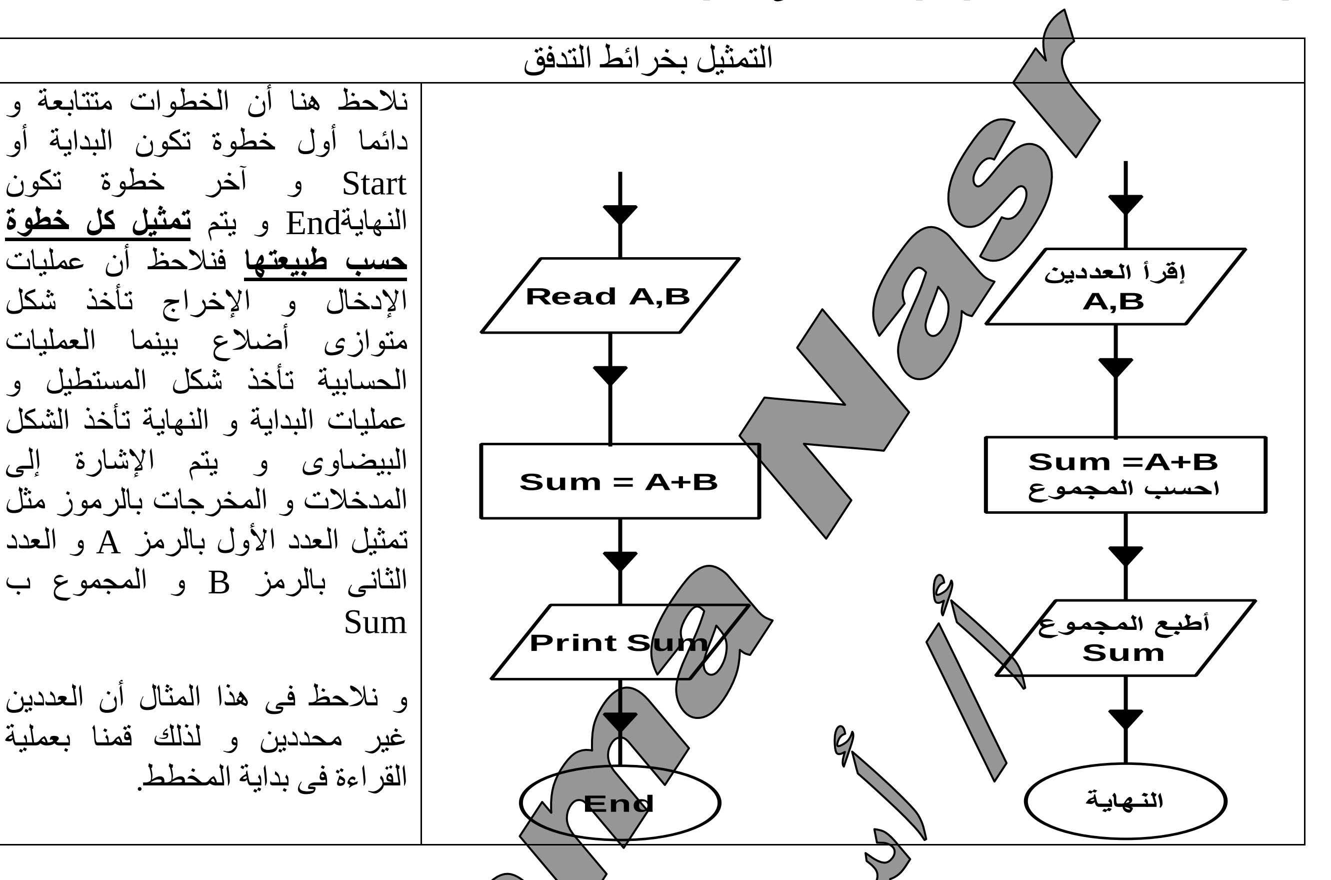
مثال:- ارسم خريطة التدفق و اكتب السودو كود لبرنامج يقوم بحساب مجموع عددين

- نفرض العدد الأول = A و العدد الثانى = B و المجموع = Sum

و لتسهيل عمل خريطة التدفق يفضل أولا كتابة الخوارزمية وهى التى تمثل خطوات حل المسألة و بمقارنة الخوارزمية بخريطة التدفق يتضح الارتباط بين الاثنين حيث الخوارزمية تعبير لفظى و خريطة التدفق تعبير بيانى لخطوات حل المسألة، و الخوارزمية تبدأ دائما بالخطوة الاولى (إبدأ) و تنتهى بخطوة (النهاية)هذه هى الخوارزمية:-

١ - ابدأ
٢ - ادخل (إقرأ) العددين الاول و الثانى.
٣ - احسب المجموع = العدد الاول + العدد الثانى
٤ - اطبع المجموع
٥ - النهاية

التمثيل بخرائط التدفق



نلاحظ هنا أن الخطوات متتابعة و دائما أول خطوة تكون البداية أو Start و آخر خطوة تكون النهايةEnd و يتم **<u>تمثيل كل خطوة حسب طبيعتها</u>** فنلاحظ أن عمليات الإدخال و الإخراج تأخذ شكل متوازى أضلاع بينما العمليات الحسابية تأخذ شكل المستطيل و عمليات البداية و النهاية تأخذ الشكل البيضاوى و يتم الإشارة إلى المدخلات و المخرجات بالرموز مثل تمثيل العدد الأول بالرمز A و العدد الثانى بالرمز B و المجموع ب Sum

و نلاحظ فى هذا المثال أن العددين غير محددين و لذلك قمنا بعملية القراءة فى بداية المخطط.

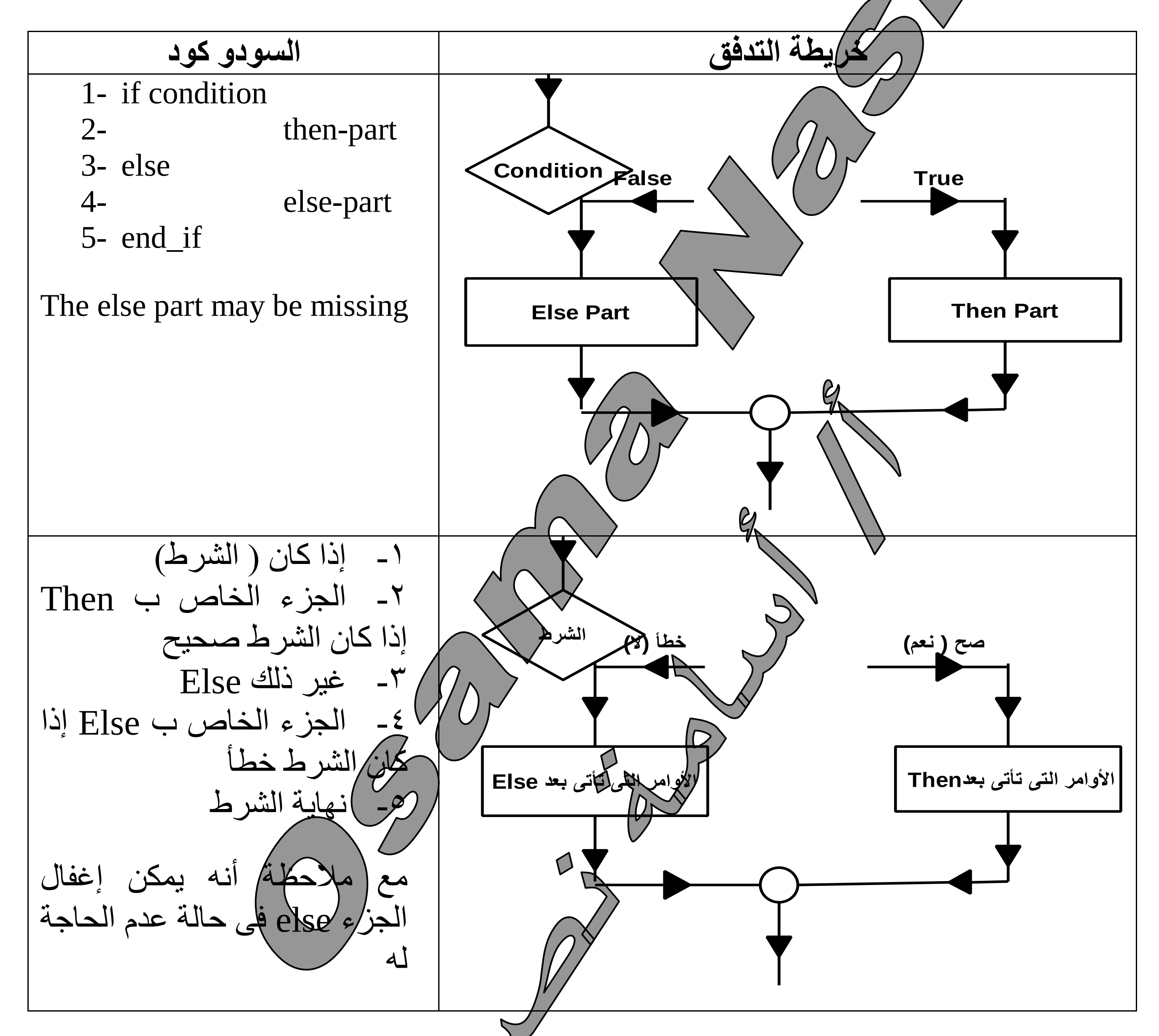
أما السودو كود فيمكن كتابته كما يلى:-

- نفرض العدد الأول = A و العدد الثانى = B و المجموع = Sum

الشرح	السودو كود
١- إقرأ العددين A و B ٢- احسب المجموع = A+B ٣- أطبع قيمة المجموع ٤- النهاية	1- Read A and B 2- sum = A+B 3- Print Sum 4- End Process

هذا الشكل من الأمثلة يمكن أن يستخدم فى أمثلة أخرى مثل حساب مساحة مربع أو دائرة أو مستطيل .... الخ أو حساب ناتج ضرب عددين أو ناتج طرح عددين .. الخ كل ما سيتم تغييره هو محتويات خطوة القراءة و المعادلة الخاصة بالحساب و محتويات خطوة الطباعة.

ثانيا :- <u>نمط الاختيار و التشعب</u> :- و فى هذا النمط يكون المطلوب اتخاذ قرار تكون نتيجته نعم أو لا (صح أو خطأ) و فى هذه الحالة تكون نعم أو صح تعبيرا عن تحقق الشرط و تكون لا أو خطأ تعنى عدم تحقق الشكل



خريطة التدفق	السودو كود
	1- if condition 2- then-part 3- else 4- else-part 5- end_if The else part may be missing
	١- إذا كان (الشرط) ٢- الجزء الخاص ب Then إذا كان الشرط صحيح ٣- غير ذلك Else ٤- الجزء الخاص ب Else إذا كان الشرط خطأ ٥- نهاية الشرط مع ملاحظة أنه يمكن إغفال الجزء else فى حالة عدم الحاجة له

مثال:- ارسم خريطة التدفق و اكتب السودو كود الخاص ببرنامج يقوم بقسمة عددين.

<u>الحل:-</u>

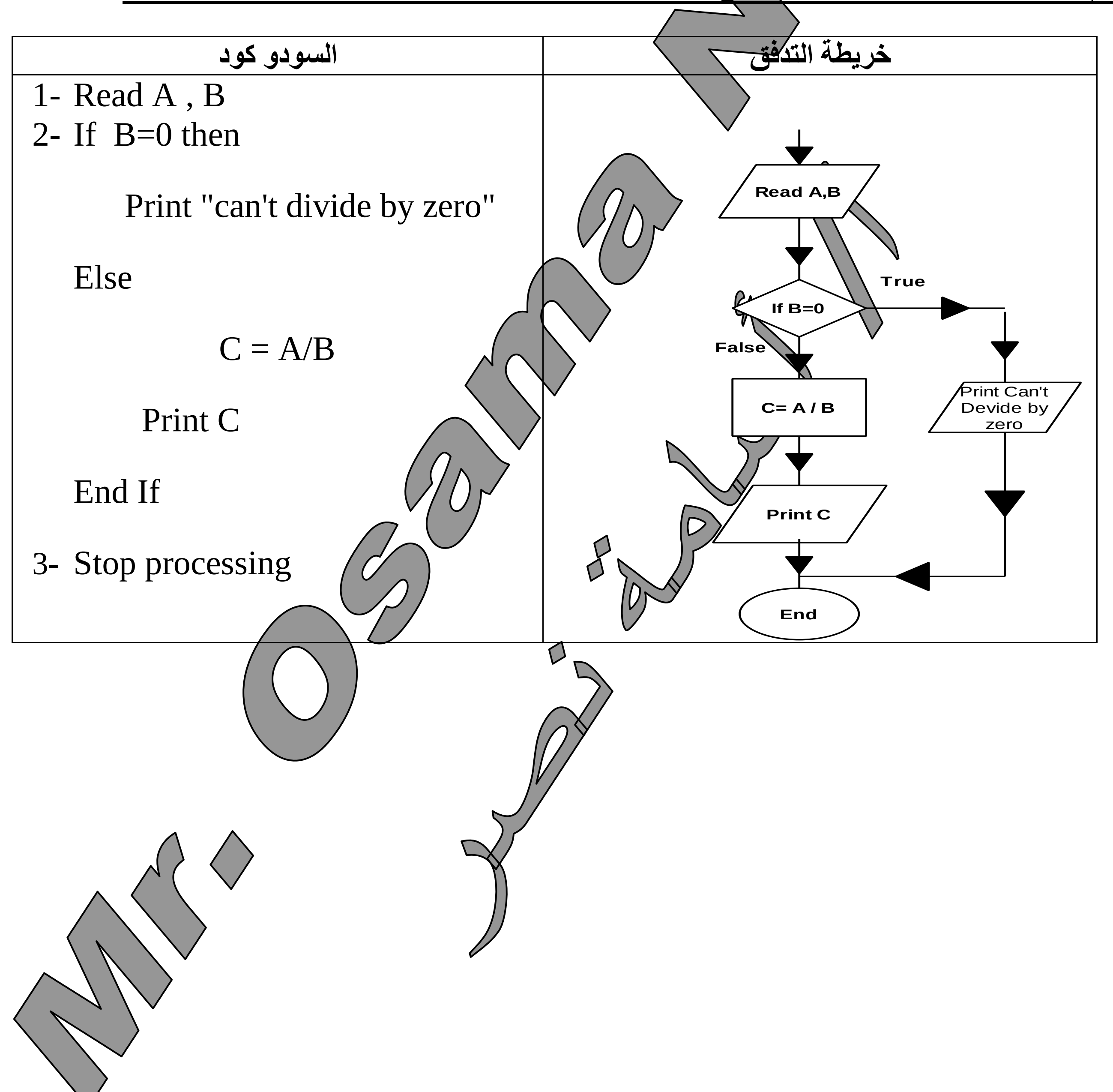
<u>قبل الحل لابد من التفكير جيدا فى كافة الشروط و الخطوات اللازمة لإنجاز عملية القسمة و عملية القسمة تتشابه مع كافة العمليات الحسابية و لكن يوجد شرط واحد فقط و هو ان لا يكون المقام يساوى صفر لأن ناتج القسمة فى هذه الحالة سيكون كمية غير محددة و هذا خطأ يجب تجنبه فى الكمبيوتر كما يتم تجنبه فى حل المسائل يدويا</u>

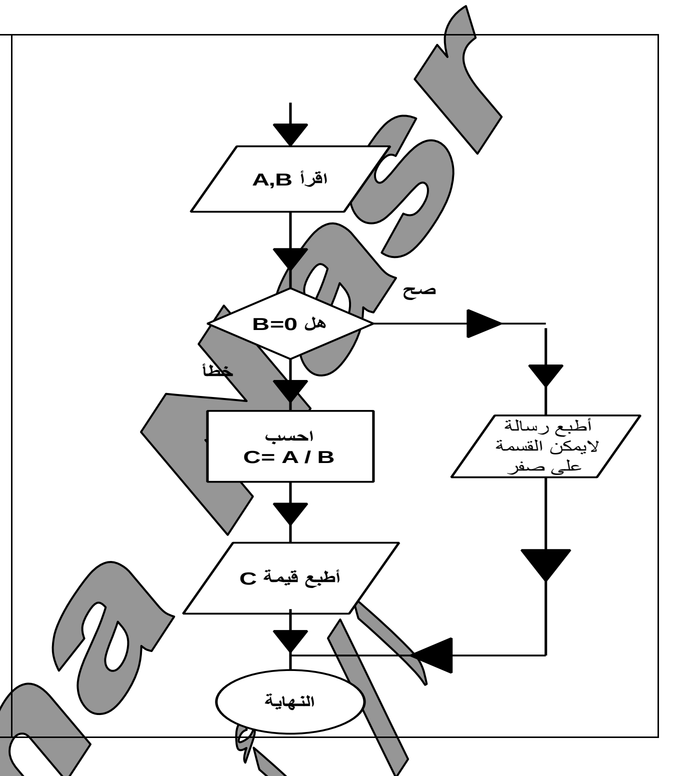
الخوارزمية:

١- ابدأ

٢- ادخل العددين الاول و الثانى
٣- اذا كان العدد الثانى يساوى صفر اطبع "لا يمكن اجراء عملية القسمة" ثم اذهب للخطوة ٦ و إذا كان العدد الثانى لا يساوى صفر إذهب للخطوة ٤
٤- حاصل القسمة = العدد الاول ÷ العدد الثانى
٥- اطبع حاصل القسمة
٦- النهاية

<u>سيم الاشارة للعدد الأول بالرمز **A** و الثانى بالرمز **B** و حاصل عملية القسمة بالرمز **C** حيث **C=A / B**</u>

خريطة التدفق	السودو كود
	1- Read A , B 2- If B=0 then Print "can't divide by zero" Else C = A/B Print C End If 3- Stop processing



١- أقرأ قيمة كلا من A, B
٢- إذا كانت B=0

أطبع رسالة " لا يمكن القسمة على صفر" ثم قم بإنهاء البرنامج

ما عدا ذلك

احسب C= A/B

أطبع قيمة C

نهاية الشرط

٣- توقف (إنهاء البرنامج)

ثالثا:- نمط التكرار

و يستخدم هذا النمط لتكرار خطوة أو مجموعة من الخطوات لعدد محدد من المرات

و هناك مفهومان يجب معرفتهما قبل الخوض فى أمثلة على التكرار و هما:-

١- العداد ٢- المجمع

أولا:- العداد

العداد هو عبارة عن متغير يأخذ قيمة ابتدائية ثم يتم زيادته دوريا بقيمة ثابته و من أمثلة ذلك:-

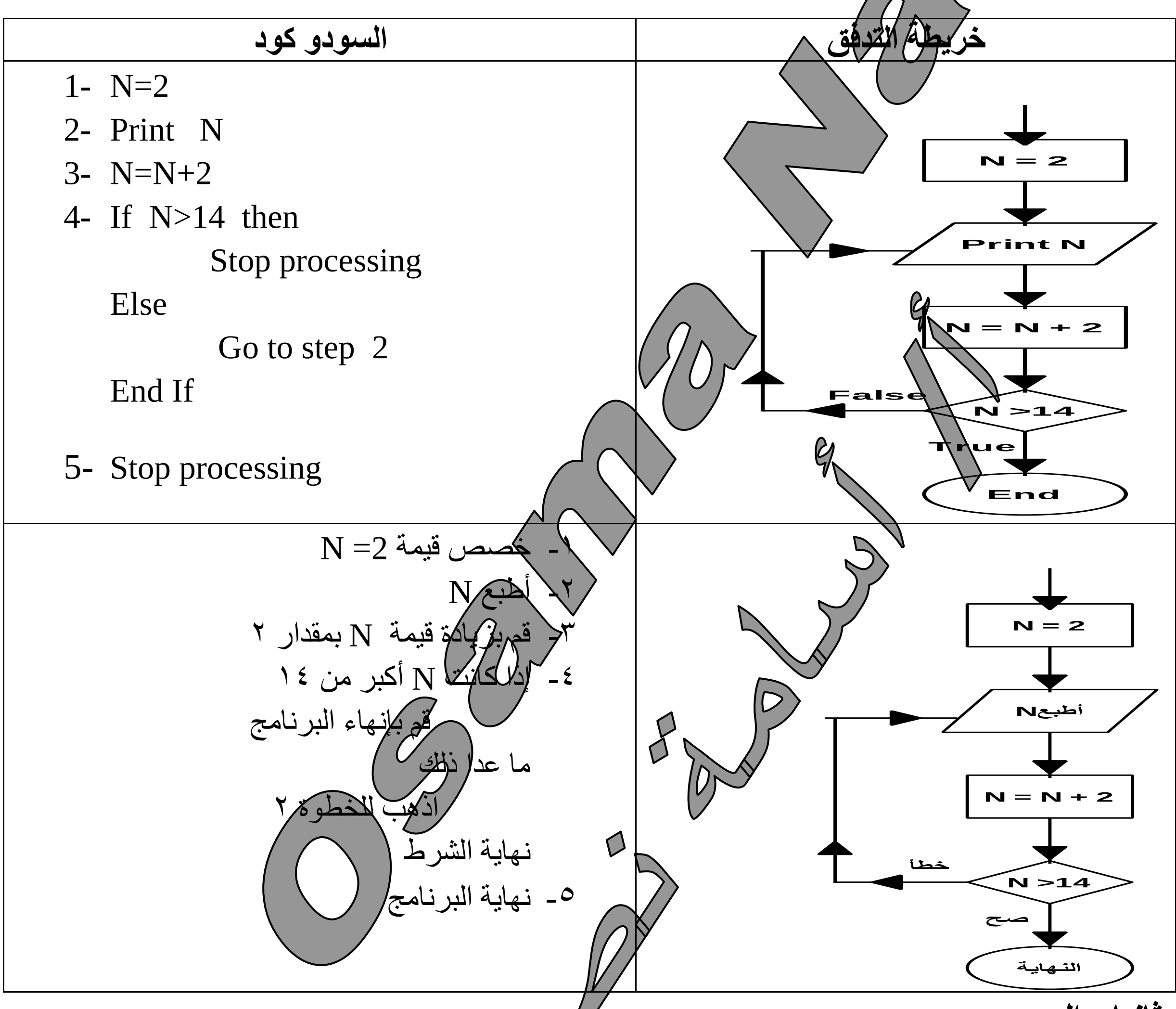
- إذا أردنا أن نصنع متغير يعطينا الأرقام الصحيحة من ١ إلى ٥ فإن القيمة المبدئية ستكون ١ و قيمة الزيادة ستكون ١ و بذلك ستكون القيم التى سيكتسبها هى ١ ، ١+١ ، ٢+١ ، ٣+١ ، ٤+١ أى ١ ، ٢ ، ٣ ، ٤ ، ٥
- إذا أردنا أن نصنع متغيرا يعطينا الأعداد الفردية من ٧ إلى ١٩ فإن القيمة المبدئية له ستكون ٧ و ستكون قيمة الزيادة ٢ (للحصول على أعداد فردية صحيحة) و بذلك ستكون القيم التى سيأخذها هى ٧ ، ٧+٢ ، ٩+٢ ، ١١+٢ ، ١٣+٢ ، ١٥+٢ ، ١٧+٢ أى ٧ ، ٩ ، ١١ ، ١٣ ، ١٥ ، ١٧ ، ١٩

مثال :- ارسم خريطة التدفق و اكتب السودو كود لبرنامج يقوم بطباعة الأعداد الزوجية من ٢ إلى ١٤

الخوارزمية:-

١- إبدأ
٢- تخصيص قيمة مبدئية للعداد(هو نفسه في هذه الحالة العدد الزوجى المطلوب طباعته)= ٢

٣- اطبع العداد
٤- زيادة قيمة العداد بمقدار ٢ = القيمة السابقة للعداد + ٢ (للحصول على ٤ ثم ٦ .... الخ)
٥- اذا كانت قيمة العداد أقل من ١٤ أو تساوى ١٤ اذهب للخطوة رقم ٣ و إذا كانت قيمة العداد أكبر من ١٤ إذهب للخطوة ٦
٦- النهاية



خريطة التدفق	السودو كود
	1- N=2 2- Print N 3- N=N+2 4- If N>14 then Stop processing Else Go to step 2 End If 5- Stop processing
	١- خصص قيمة N =2 ٢- أطبع N ٣- قم بزيادة قيمة N بمقدار ٢ ٤- إذا كانت N أكبر من ١٤ قم بإنهاء البرنامج ما عدا ذلك اذهب للخطوة ٢ نهاية الشرط ٥- نهاية البرنامج

ثانيا : المجمع

المجمع هو متغير يأخذ قيمة ابتدائية عادة ما تكون الصفر باعتباره عنصر محايد جمعى ثم يتم إضافة قيمة متغير آخر له بشكل متكرر بعد زيادته بشكل متكرر أيضا .

و لنأخذ المثال السابق و نطوره بحيث يقوم بطباعة ***مجموع الأعداد الزوجية من ٢ إلى ١٤***.

الخوارزمية:-

١- إبدأ
٢- العداد (هو نفسه العدد)= ٢
٣- قيمة المجموع = صفر (عنصر محايد جمعى)
٤- المجموع = المجموع القديم + العداد (قيمة العدد)
٥- العداد = العداد القديم + ٢

٦- اذا كانت قيمة العداد أقل من أو تساوى ١٤ اذهب للخطوة ٤ و إذا كانت قيمة العداد أكبر من ١٤ إذهب للخطوة ٧
٧- اطبع المجموع
٨- النهاية

خريطة التدفق	السودو كود
	1- N=2 2- Sum =0 3- Sum=Sum + N 4- N=N+2 5- If N > 14 then Print Sum Stop processing Else Go to Step 3 End If 6- Stop processing
	١- القيمة المبدئية لـ N=2 ٢- القيمة المبدئية لـ Sum= 0 ٣- المجموع = المجموع السابق + قيمة N Sum = Sum+N ٤- قم بزيادة قيمة N بمقدار ٢ N=N+2 ٥- إذا كانت قيمة N أكبر من ١٤ أطبع قيمة المجموع Sum أوقف البرنامج ٦- ما عدا ذلك اذهب للخطوة رقم ٣ ٧- نهاية الشرط ٨- نهاية البرنامج

و نلاحظ فى المثال السابق أنه تم استخدام متغيرين أولهما هو المتغير N و الذى تم استخدامه لتوليد الأعداد من ٢ إلى ١٤ و المتغير الآخر هو Sum و تم استخدامه لحساب المجموع التراكمى للأعداد الزوجية و تم تخصيص قيمة مبدئية له تساوى صفر باعتبار الصفر عنصر محايد جمعى و لن يؤثر فى النتيجة.

و على نمط هذا البرنامج يمكن عمل برنامج لإيجاد مجموع الأعداد الفردية من ١ إلى ٢٧ مثلا أو مجموع الأعداد من ١ إلى ١٠٠ و هكذا و كل ما سيتغير هو ثلاثة خطوات أولها الخطوة التى تحدد القيمة المبدئية (الخطوة رقم ٢ فى الرسم و ١ فى السودو كود) و خطوة قيمة الزيادة (الخطوة ٥ فى الرسم و ٥ فى السودو كود) و خطوة الشرط (خطوة ٦ فى الرسم و ٦ فى السودو كود)

ملاحظة:- نلاحظ فى الرسم السابق فى الخطوة رقم ٢ أنه تم دمج خطوتين متشابهتين معا و هذا أمر ممكن طالما تشابهت الخطوات و تتابعت أو يمكن فصلها كما تم كتابته فى السودو كود و كلا الأمرين صحيح.



أمثلة إضافية

مثال ١ :- أرسم خريطة تدفق لبرنامج يقوم بحساب مساحتى مستطيل و دائرة



الرموز المستخدمة

طول المستطيل = A

عرض المستطيل = B

مساحة المستطيل = R

نصف قطر الدائرة = C

مساحة الدائرة = S

السودوكود

1-Read A,B,C
2-R = A * B
3-S = 22/7 * C^2
4-Print R , S
5-Stop

مثال ٢ :- ارسم خريطة تدفق لبرنامج يقوم بتمييز الاعداد الموجبة عن الاعداد السالبة مع مراعاة حالة الرقم (صفر)

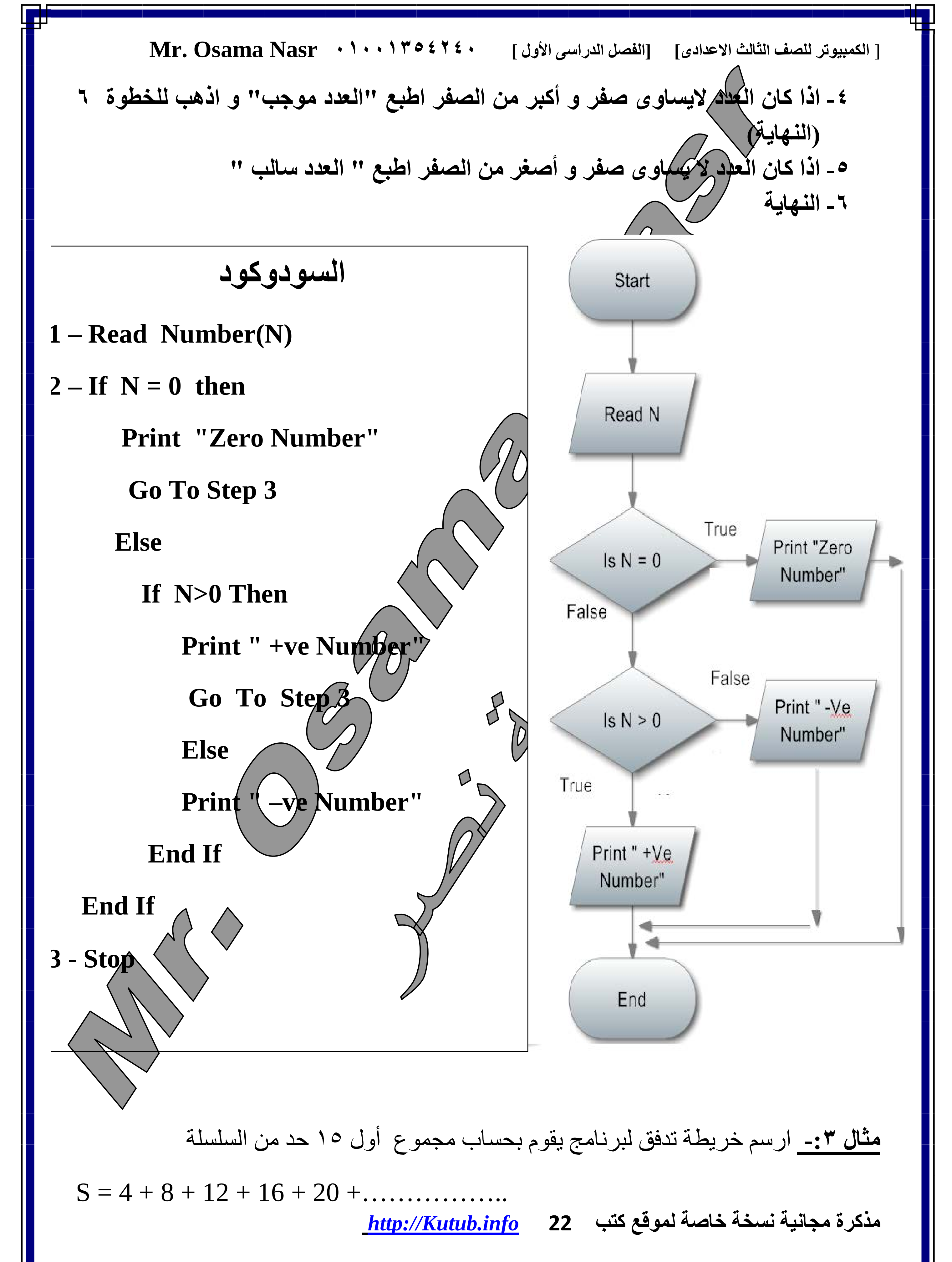
الخوارزمية:-

١ - إبدأ
٢ - ادخل (إقرأ)العدد
٣ - اذا كان العدد يساوى صفر اطبع "العدد هو صفر" و اذهب للخطوة رقم ٦ (النهاية)

٤- اذا كان العدد لايساوى صفر و أكبر من الصفر اطبع "العدد موجب" و اذهب للخطوة ٦ (النهاية)

٥- اذا كان العدد لا يساوى صفر و أصغر من الصفر اطبع " العدد سالب "

٦- النهاية



السودوكود

1 – Read Number(N)

2 – If N = 0 then

Print "Zero Number"

Go To Step 3

Else

If N>0 Then

Print " +ve Number"

Go To Step 3

Else

Print " –ve Number"

End If

End If

3 - Stop

مثال ٣:- ارسم خريطة تدفق لبرنامج يقوم بحساب مجموع أول ١٥ حد من السلسلة

S = 4 + 8 + 12 + 16 + 20 +……………..

الخوارزمية :-

١- ابدأ
٢- العدد = ٤ (القيمة المبدئية)
٣- العداد = صفر
٤- المجموع = صفر
٥- المجموع = المجموع + العدد
٦- العداد = العداد + ١
٧- العدد = العدد + ٤ (قيمة الزيادة بين الحدود)
٨- اذا كان العداد أقل من ١٥ اذهب للخطوة رقم ٥
٩- اذا كان العداد أكبر من أو يساوى ١٥ اطبع المجموع
١٠- النهاية

الرموز المستخدمة

العدد (يمثل الحد) = N ، العداد = C

المجموع = Sum

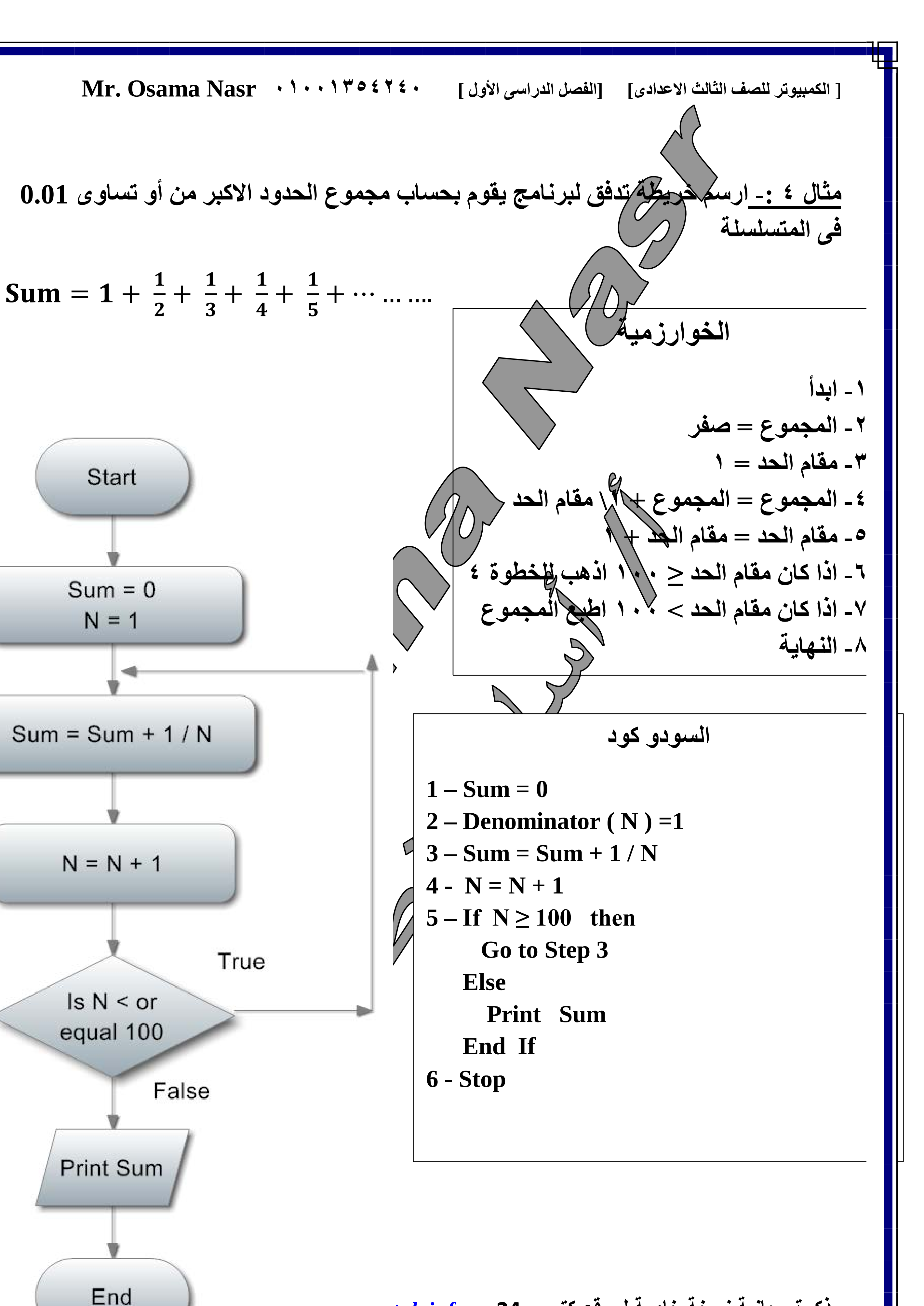
Start
N = 4
C = 0
Sum = 0
Sum = Sum + N
C = C + 1
N = N + 4
Is C < 15
True
False
Print Sum
End

السودو كود

```
1- Number ( N) = 4
2- Counter ( C ) = 0
3- Summmation(Sum)= 0
4- Sum = Sum + N
5- C = C + 1
6- N = N + 4
7- If  C < 15 then
        Go to Step 4
   Else
        Print Sum
   End If
8- Stop
```

مثال ٤ :- ارسم خريطة تدفق لبرنامج يقوم بحساب مجموع الحدود الاكبر من أو تساوى 0.01 فى المتسلسلة

$$\mathbf{Sum} = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \cdots \ldots \ldots$$



الخوارزمية

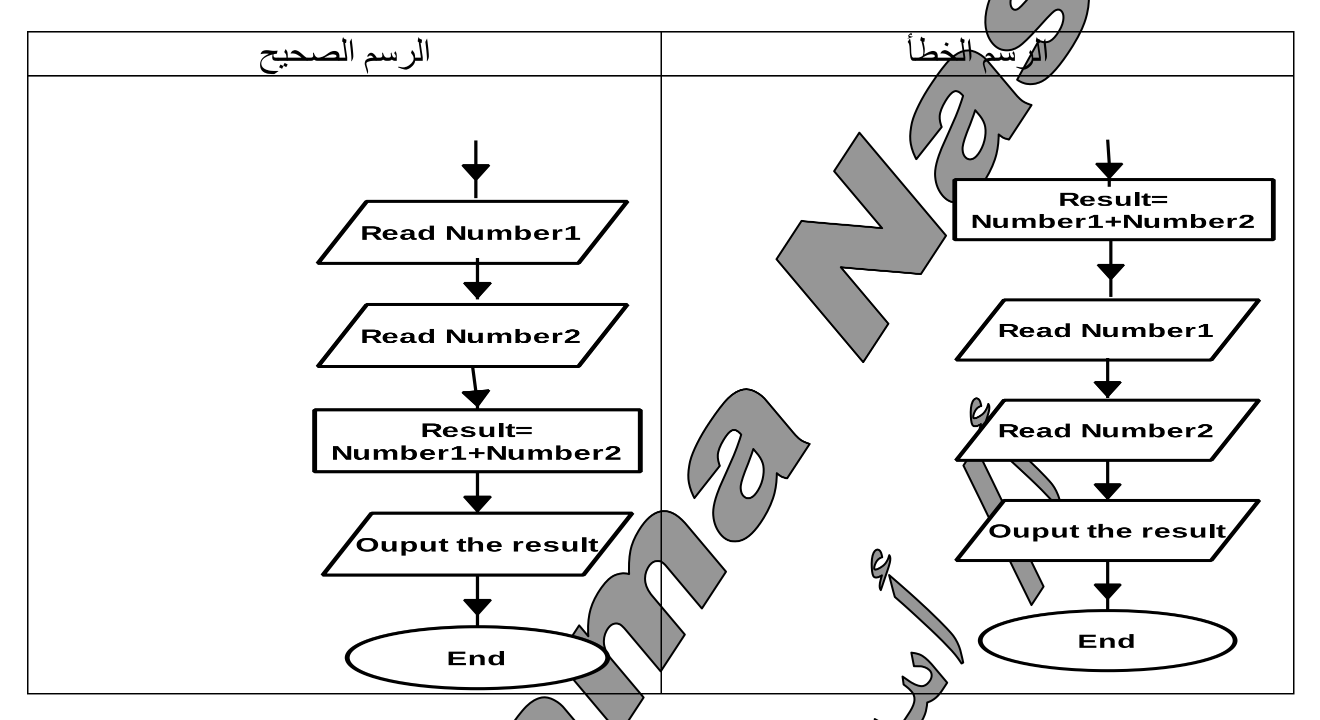
١ـ ابدأ
٢ـ المجموع = صفر
٣ـ مقام الحد = ١
٤ـ المجموع = المجموع + ١/ مقام الحد
٥ـ مقام الحد = مقام الحد + ١
٦ـ اذا كان مقام الحد ≤ ١٠٠ اذهب للخطوة ٤
٧ـ اذا كان مقام الحد > ١٠٠ اطبع المجموع
٨ـ النهاية

السودو كود

1 – Sum = 0
2 – Denominator (N) =1
3 – Sum = Sum + 1 / N
4 - N = N + 1
5 – If N ≥ 100 then
Go to Step 3
Else
Print Sum
End If
6 - Stop

حل أسئلة الفصل الثانى

س١- قامت دنيا برسم خريطة تدفق لعملية جمع عددين و إظهار الناتج كالتالى

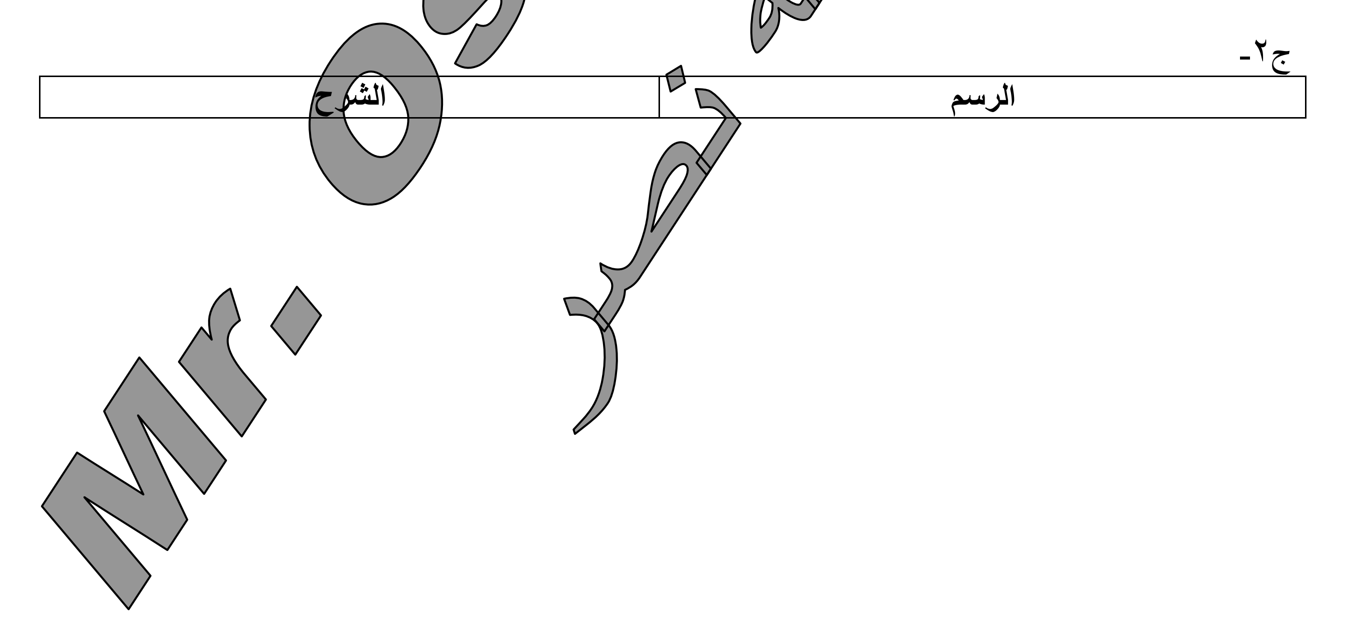
الرسم الخطأ	الرسم الصحيح
	

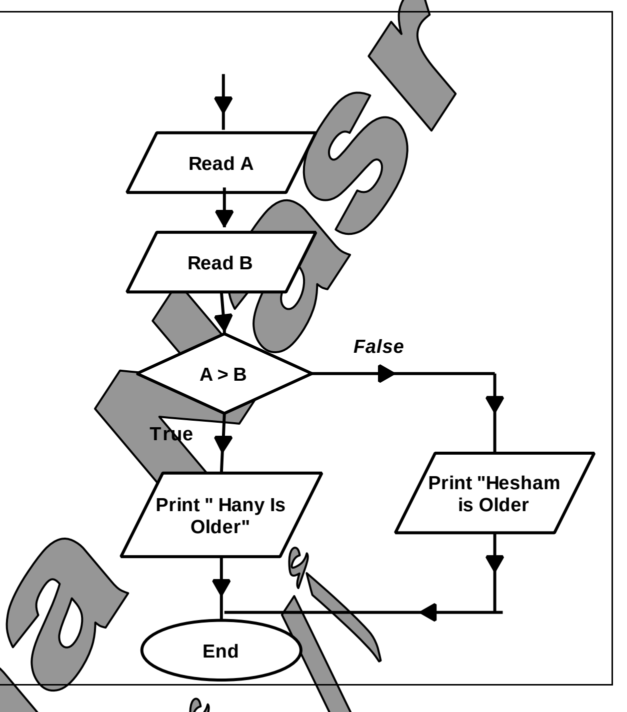
الشرح: كانت خطوة حساب المجموع Result سابقة على قراءة العددين و بالتالى تكون النتيجة غير صحيحة. و لذلك تم تقديم خطوات القراءة قبل حساب المجموع.

س٢- أرسم خريطة تدفق و أكتب السودو كود لقراءة عمرى هانى و هشام ثم كتابة اسم أكبرهما.

ج٢-

الرسم	الشرح



	عمر هانى = A عمر هشام = B مقارنة عمر هانى بعمر هشام. هل عمر هانى (A) أكبر من عمر هشام (B) إذا كانت النتيجة صحيحة اطبع عمر هانى أكبر. إذا كانت النتيجة خطأ أطبع عمر هشام أكبر. **ملاحظة هامة:-** نفس الخريطة يمكن أن تستخدم لمقارنة عددين و تحديد أيهما أكبر. أو تحديد الأعداد الموجبة و السالبة .... الخ

- نفرض أن عمر هانى = A عمر هشام = B

السودو كود	الشرح
1- Read A , B 2- If A >B Then Print "Hany" , Stop processing Else Print "Hesham" End If 3- Stop processing	١- اقرأ عمر هانى A ٢- اقرأ عمر هشام B ٣- إذا كان عمر هانى A أكبر من عمر هشام B إذاً أطبع " هانى" ، قم بإنهاء البرنامج ما عدا ذلك أطبع " هشام" نهاية الشرط ٤- قم بإنهاء البرنامج

س٣- ارسم خريطة تدفق و أكتب السودو كود لعملية قراءة درجة الحرارة بالفهرنهايت °F ثم تحويلها إلى ما يقابلها بالدرجات المئوية °C ثم قم بطباعة درجة الحرارة بالفهرنهايت و بالدرجات المئوية، مع العلم أن العلاقة بينهما كالآتى:-

$°C=(5/9) \times (°F - 32)$

نفرض الدرجة بالفهرنهايت = F و الدرجة المئوية = C



الرسم	الشرح
Read F → C=(5/9) x(F-32) → Print F,C → End	١- قراءة الدرجة بالفهرنهايت F° ٢- يتم حساب درجة الحرارة بالنظام المئوى C° حسب المعادلة ٣- طباعة الدرجة بالفهرنهايت و بالنظام المئوى ٤- النهاية

السودو كود	الشرح
1- Read F 2- C =(5/9) x (F-32) 3- Print F,C 4- End Process	١- اقرأ قيمة الدرجة بالفهرنهايت F ٢- أحسب القيمة المكافئة بالدرجات المئوية C=(5/9) x (F-32) ٣- أطبع الدرجة بالفهرنهايت F و الدرجة المئوية C ٤- نهاية البرنامج

السؤال الرابع:- أرسم الشكل المناسب فى الفراغات التالية.

أ- يستخدم الشكل .... لبدء و إنهاء لوحة التدفق.

ب – للمقارنة بين عددين نستخدم الشكل

جـ- يستخدم الشكل .... للإدخال و الإخراج.

د- للمعالجة نستخدم الشكل

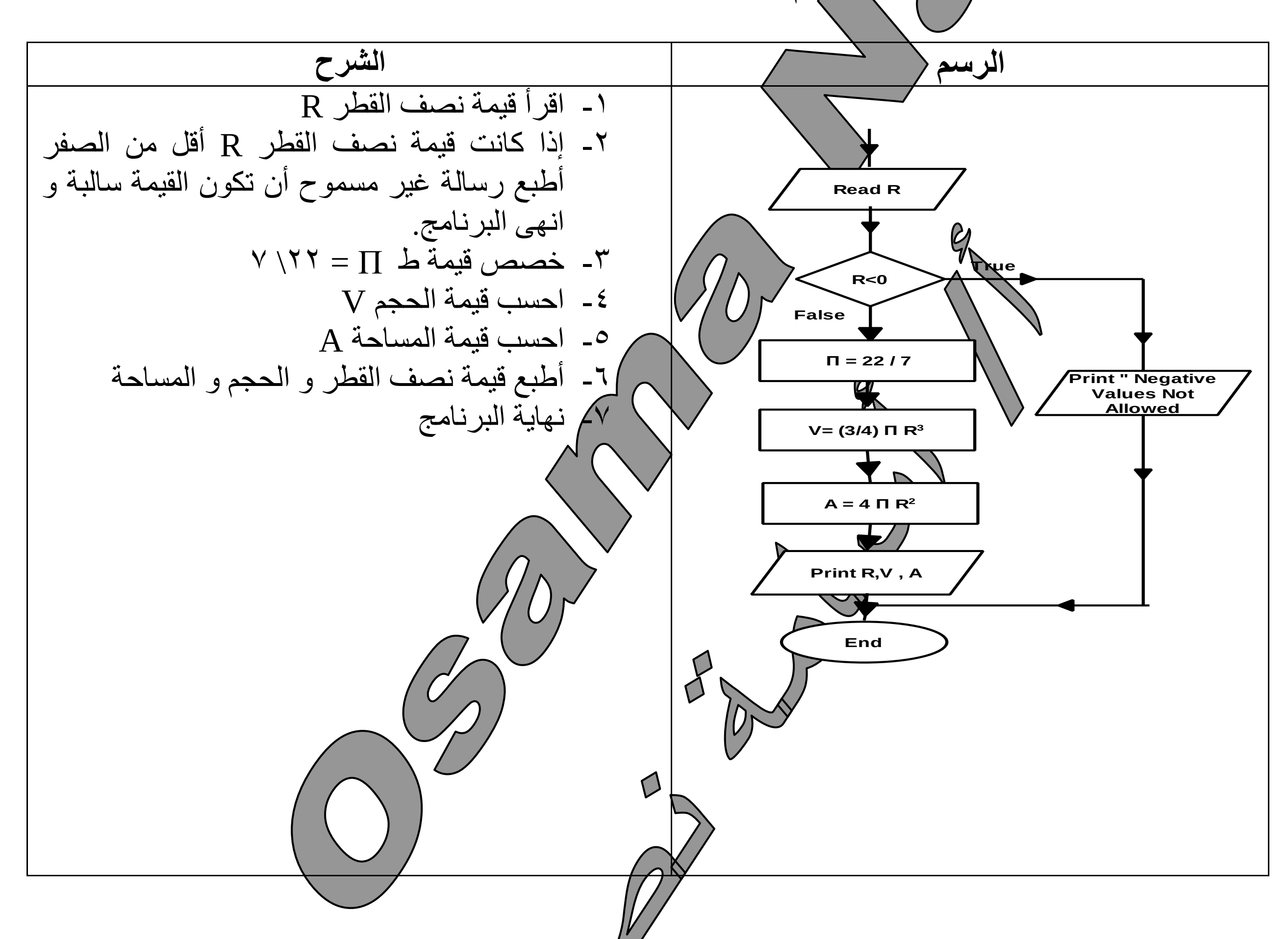
هـ- يستخدم الشكل .... لتوصيل شكل بآخر.

السؤال الخامس:- ارسم خريطة التدفق و أكتب السودو كود لعملية قراءة نصف قطر كرة r ثم ايجاد حجم الكرة V و مساحة سطحها A علما بأن المعادلات المطلوبة هى:

$V= (3/4) \Pi r^3$, $A = 4 \Pi r^2$, Π=ط= 22 / 7

إذا كان نصف القطر سالب ، يجب طباعة رسالة تحذير ثم إنهاء البرنامج.

نفرض نصف قطر الكرة = R و الحجم = V و المساحة = A

الشرح	الرسم
١- اقرأ قيمة نصف القطر R ٢- إذا كانت قيمة نصف القطر R أقل من الصفر أطبع رسالة غير مسموح أن تكون القيمة سالبة و انهى البرنامج. ٣- خصص قيمة ط Π = ٢٢\ ٧ ٤- احسب قيمة الحجم V ٥- احسب قيمة المساحة A ٦- أطبع قيمة نصف القطر و الحجم و المساحة ٧- نهاية البرنامج	

الشرح	السودو كود
١- أقرأ نصف قطر الكرةR ٢- إذا كانت قيمة R سالبة أطبع " قيمة نصف القطر لابد أن تكون موجبة" ، قم بإنهاء البرنامج ما عدا ذلك قيمة ط = Pi = ٢٢÷٧ أحسب الحجم V=(3/4) * Pi * R3 أحسب المساحة	1- Read R 2- If R < 0 then Print "the value of radius Should be Positive" , End Process Else Let Pi -22/7 $V=(3/4) * Pi * R^3$ $A= 4*Pi * R^2$



A= 4*Pi * R2 أطبع نصف القطر و الحجم و المساحة نهاية الشرط ٣- نهاية البرنامج	End If 3- Print R , V , A 4- Stop processing

حل أسئلة الفصل الثالث

أجابة السؤال الأول

الشرح	السودو كود
١- اقرأ العددين الأول n1 و الثانى n2 ٢- اقرأ العملية الحسابية ٣- إذا كانت العملية الحسابية هى الجمع إذاً النتيجة Result =(n1+n2) غير ذلك إذا كانت العملية الحسابية هى الطرح إذاً النتيجة Result =(n1 -n2) غير ذلك إذاً كانت العملية الحسابية هى الضرب إذاً النتيجة Result =(n1 * n2) غير ذلك إذا كانت العملية الحسابية هى القسمة إذاً النتيجة Result =(n1 / n2) غير ذلك اطبع " عملية خطأ" ثم قم اذهب للخطوة ٥ نهاية الشرط نهاية الشرط نهاية الشرط نهاية الشرط ٤- اطبع النتيجة (Result) ٥- نهاية البرنامج	1- Read n1 , n2 2- Read the arithmetic operation 3- If operation is addition then Result= n1+ n2 Else If operation is subtraction then Result = n1- n2 Else If operation is multiplication then Result= n1* n2 Else If operation is Division then Result= n1 / n2 Else Print "Wrong operation" and go to step 5 End If End If End If End If 4- Print Result 5- End Process

باقي الأسئلة تم اجابتها مع أسئلة الفصل الثاني

الفصل الرابع

مقدمة عن البرمجة الشيئة

Introduction To Object Oriented Programming

بخلاف لغات البرمجة القديمة يستخدم Visual Basic.net أسلوب البرمجة بالكائنات (Objects) و لكل كائن خصائص (Properties) و وسائل(وظيفة مرتبطة بوقوع حدث) و تسمى(Methods) و أحداث تقع للكائن تسمى(Events) و كل Object له فئة معينة تسمى Class

الكائن:-

هو شيء له وجود (يمكن اثبات وجودة) و من أمثلة الكائنات السيارة – القلم الرصاص – القلم الجاف – الباب- الشباك .... الخ و يلاحظ أن بعض الكائنات الأخرى مثل السيارة تحتوى على كائنات أخرى داخلها مثل العجل – الأبواب – الزجاج – الموتور ... الخ

و فى لغة البرمجة الشيئية Object Oriented Programming OOP يمكن اعتبار النافذة Windows

و صناديق الحوار(الرسائل التى تظهر للمستخدم و تطلب منه إجابة) و تسمى Dialog Boxes أمثلة للكائنات.

و على هذا يمكن تعريف الكائن كالآتى:-

الكائن(Object) هو شئ له وجود حيث نتعامل مع الكائنات(Objects) بصورة دائمة فى حياتنا اليومية و الكائن الواحد (Object) يمكن أن يتكون من عدد من الكائنات(Objects).

الخصائص:-

لكل كائن(Object) خصائص(Properties) معينه تميزه و تحدد ملامحه مثل أسمه، حجمه ، لونه، طوله، عرضه، حجمه ، وزنه ... الخ

مثال ذلك:-

Object الكائن	Property الخاصية	Value قيمة الخاصية
القلم	الاسم	قلم
	اللون	أزرق
	الطول	١٢ سم
	الوزن	٣٥ جرام
السيارة	الاسم	سيارة
	اللون	ابيض
	عدد الأبواب	٢
	عدد مقاعد الركاب	٥
	عدد العجلات	٤

و يمكن التعبير عن ذلك من خلال لغة البرمجة التى سندرسها لاحقا و التى من خلالها كتابة خاصية الكائن كالآتى:-

ObjectName.Property=Value

قيمة = الخاصية . اسم الكائن

و على سبيل المثال إذا كان لدينا كائن اسمه MyCar و هو السيارة فى الجدول أعلاه فإن خاصية لون السيارة يمكن التعبير عنها كالاتى:-

MyCar.Color=White

و هنا يكون الكائن هو MyCar أما الخاصية فهى Color و قيمتها White

و يمكن تعريف الخاصية (Property):

تحدد شكل و سمة الكائن و كل كائن له مجموعة من الخصائص

الوسائل : Methods

هى الوظائف و الأفعال التى تكون مصاحبة للكائن (Object). أى هى أفعال محددة مسبق لكل كائن.

مثال ذلك السيارة MyCar لونها أبيض(خاصية) و تسير فى الطريق(وسيلة)

و يتم التعامل مع ذلك من خلال لغة البرمجة كالاتى

ObjectName.Method

الوسيلة . اسم الكائن

و حيث أن السيارة يمكن أن تبدأ السير و يمكن أن تتوقف يمكن التعبير عن الوسيلتين(الوظيفتين) كالتالى:

MyCar.Start()   MyCar.Stop()

الأحداث Events :-

كل كائن(Object) له أحداث معينة(Events) تقع عليه و يستجيب لها، حيث يمكن للمبرمج وضع مجموعة معينة من الأوامر التى يتم تنفيذها عند وقوع حدث معين.

إذا الحدث هو عبارة عن الفعل الذى يقع على الكائن و يستجيب له.

فإذا افترضنا أن لدينا نفس الكائن السابق و هو السيارة و تم الضغط على دواسة الوقود(البنزين) فإن السيارة تزيد سرعتها أكثر و أكثر كلما ضغطنا على دواسة(البنزين)

و على هذا فإن لدينا

كائن(Object):- سيارة الحدث(Event):- الضغط على دواسة الوقود

الوسيلة(Methods):- تسير – تسرع

و هنا الحدث هو الضغط على دواسة الوقود و الاستجابة للحدث تتمثل فى زيادة سرعة السيارة.

و إذا افترضنا حدث آخر و هو على سبيل المثال :- الضغط على دواسة الفرامل فإن الوسيلة (Method) أو الاستجابة للحدث ستتمثل فى تباطؤ سرعة السيارة تدريجيا حتى وقوفها تماما إذا استمر الضغط .

حل التمرين صفحة ٧٢

أكمل مكان النقط باختيارك الكلمة المناسبة من بين الكلمات الآتية:-

كائن – خاصية – وسيلة – حدث

✓ يعتبر جهاز الكاسيت	كائن
✓ تشغيل جهاز الكاسيت يعتبر	وسيلة
✓ إيقاف جهاز الكاسيت يعتبر	وسيلة
✓ يمكن اعتبار الإيقاف المؤقت لشريط الكاسيت	وسيلة
✓ الصوت الذى نسمعه من جهاز الكاسيت يعتبر	حدث
✓ لون جهاز الكاسيت يعتبر	خاصية
✓ حجم جهاز الكاسيت يعتبر	خاصية

مفهوم التصنيف أو الفئة (Class)

الفئة (التصنيف)هى عبارة عن قالب أو مخطط يتم منه إنشاء كائن معين، و هو يمثل جميع الخصائص و الوظائف التى سوف يحتويها الكائن بعد ذلك. أما الكائن فهو يمثل وحدة مستقلة تم إنشاؤها من الفئة Class و هى التى تستخدم فعلا لأداء الوظائف المختلفة للفئة.

أى باختصار الفئة هى مخطط الكائن و الكائن هو وحدة مستقلة تم أنشاؤها من الفئة.

من أمثلة الفئات : الرسم التصميمى لعمارة سكنية أو التصميم الهندسى لسيارة كلاهما يعتبران Class ينتج عن تنفيذهما كائنين هما العمارة السكنية و السيارة.

من كل ما سبق نستنتج أن:

البرمجة الشيئية (البرمجة بالكائنات): يقسم البرنامج فيها إلى وحدات تسمى كائنات (Objects) و لكل كائن خصائصه و وظائفه التى تميزه و التى تحدد بعد ذلك سلوكه، و تعتمد البرمجة الحديثة بشكل أساسى على هذا

النوع من البرمجة فمثلا نلاحظ أن النافذة Form و التى يتم إنشائها فى البرنامج تمثل كائن Object و هذا الكائن له خصائص تميزه مثل اللون و العرض و الارتفاع و لون الكتابة إلى غير ذلك من الخصائص، كذلك للنافذة Form وظائف مثل وظيفة إغلاق النافذة Close أو إظهارها Show أو إخفائها Hide كذلك كل شئ يتم وضعه على النافذة يمثل كائن Object له خصائص و وظائف

بعض المصطلحات الهامة فى البرمجة الشيئية

- ***مفهوم التغليف*** ***Encapsulation*** : يعنى إخفاء بيانات داخل الكائن بحيث لا يتم الوصول إليها إلا بصلاحيات معينة. مثال ذلك الدواء الذى يغلف بكبسولة حتى يذوب فقط فى المعدة أو الأمعاء أو السيارة التى لا يستطيع قيادتها إلا صاحبها الذى يمتلك مفتاحها و هو بذلك الوحيد الذى يستطيع فتحها و تشغيلها ثم قيادتها .

- ***مفهوم التوريث*** ***Inheritance*** : يقصد بالتوريث أن التصنيف –الفئة- Class ترث خصائص و وظائف فئة أخرى و هذه الفئة الجديدة تكون لديها جميع خصائص الفئة الأساسية بالإضافة إلى خصائصه الإضافية. و أبسط الأمثلة على ذلك الابن الذى يرث الصفات الأساسية من أبويه مع ملاحظة أنه و بشكل غير ظاهر يرث منهما أيضا وظائفه الأساسية.

الأسئلة:-

السؤال الأول :- ضع علامة ✓ أمام العبارة الصحيحة و علامة ✗ أما العبارة الخطأ:

١- يتم إنشاء التصنيف –الفئة- (Class)من الكائن Object (✗)
٢- يعتبر جهاز الكمبيوتر كائن object مكون من عدة كائنات (✓)
٣- يعتبر لون القلم أحد وظائف Methods القلم (✗)
٤- تشغيل جهاز الكاسيت يعتبر حدثا Event (✗)
٥- خصائص الكائن Properties تحدد شكل و سمة الكائن (✓)
٦- يتميز الكائن بأن له خصائص Properties و وظائف Methods و أحداث Events (✓)
٧- يتم إنشاء كائن Object من التصنيف Class (✓)
٨- التغليف Encapsulation يستخدم لإخفاء بيانات التصنيف Class (✓)
٩- عندما يرث تصنيف Class تصنيفا آخر فإنه يرث الخصائص فقط (✗)
١٠- من أهداف التغليف Encapsulation حماية بيانات التصنيف Class (✓)

السؤال الثانى:- أكمل مكان النقط مستخدما الكلمات الآتية لجعل الجملة صحيحة:

Object – Event – Property –Method – Class – Encapsulation – Inheritance

١- يعتبر الضغط بمؤشر الفأرة داخل النافذة **Event**
٢- يعتبر عرض النافذة **Property**
٣- الضغط على مفتاح من لوحة المفاتيح داخل النافذة يعتبر **Event**
٤- الخصائص و الوظائف التى يأخذها الابن من أبيه تعتبر **Inheritance**
٥- الذهاب إلى المدرسة يعتبر **Method**
٦- تعتبر كمية المياه الموجودة فى بحيرة **Property**
٧- الرسم التخطيطى لجهاز التسجيل الموجود بالكتالوج يعتبر **Class**
٨- يعتبر **Encapsulation** إخفاء للبيانات داخل التصنيف Class

السؤال الثالث: أكمل الجدول التالى

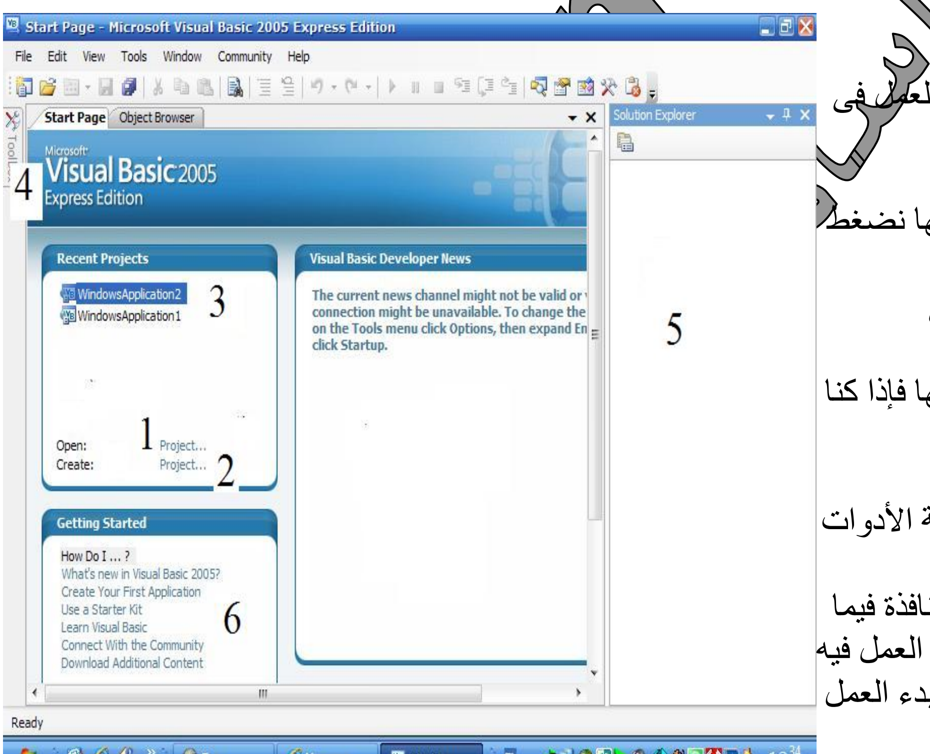
المصطلح	المقصود به
Class	الفئة (التصنيف)هى عبارة عن قالب أو مخطط يتم منه إنشاء كائن معين، و هو يمثل جميع الخصائص و الوظائف التى سوف يحتويها الكائن بعد ذلك. أما الكائن فهو يمثل وحدة مستقلة تم إنشاؤها من الفئة Class و هى التى تستخدم فعلا لأداء الوظائف المختلفة للفئة.
Property	تحدد شكل و سمة الكائن و كل كائن له مجموعة من الخصائص
Method	هى الوظائف و الأفعال التى تكون مصاحبة للكائن (Object). أى هى أفعال محددة مسبق لكل كائن.
Event	إذا الحدث هو عبارة عن الفعل الذى يقع على الكائن و يستجيب له
Object	الكائن(Object) هو شئ له وجود حيث نتعامل مع الكائنات(Objects) بصورة دائمة فى حياتنا اليومية و الكائن الواحد (Object) يمكن أن يتكون من عدد من الكائنات(Objects).
Encapsulation	يعنى إخفاء بيانات داخل الكائن بحيث لا يتم الوصول إليها إلا بصلاحيات معينة.
Inheritance	يقصد بالتوريث أن التصنيف -الفئة- Class ترث خصائص و وظائف فئة أخرى و هذه الفئة الجديدة تكون لديها جميع خصائص الفئة الأساسية بالإضافة إلى خصائصه الإضافية.

الفصل الخامس : بيئة التطوير المتكامله

Integrated Development Environment (IDE)

بيئة التطوير المتكاملة تمثلها شاشة كتابة البرامج Visual Basic .net IDE التى يستطيع المبرمج من خلالها كتابة و اختبار مشروعات الفيجوال بيسك VB Projects و يتيح IDE للمبرمج استعمال مجموعة متكاملة من الأدوات و التسهيلات فى كل مراحل كتابة VB Projects . و تتضمن هذه الأدوات ما يلى

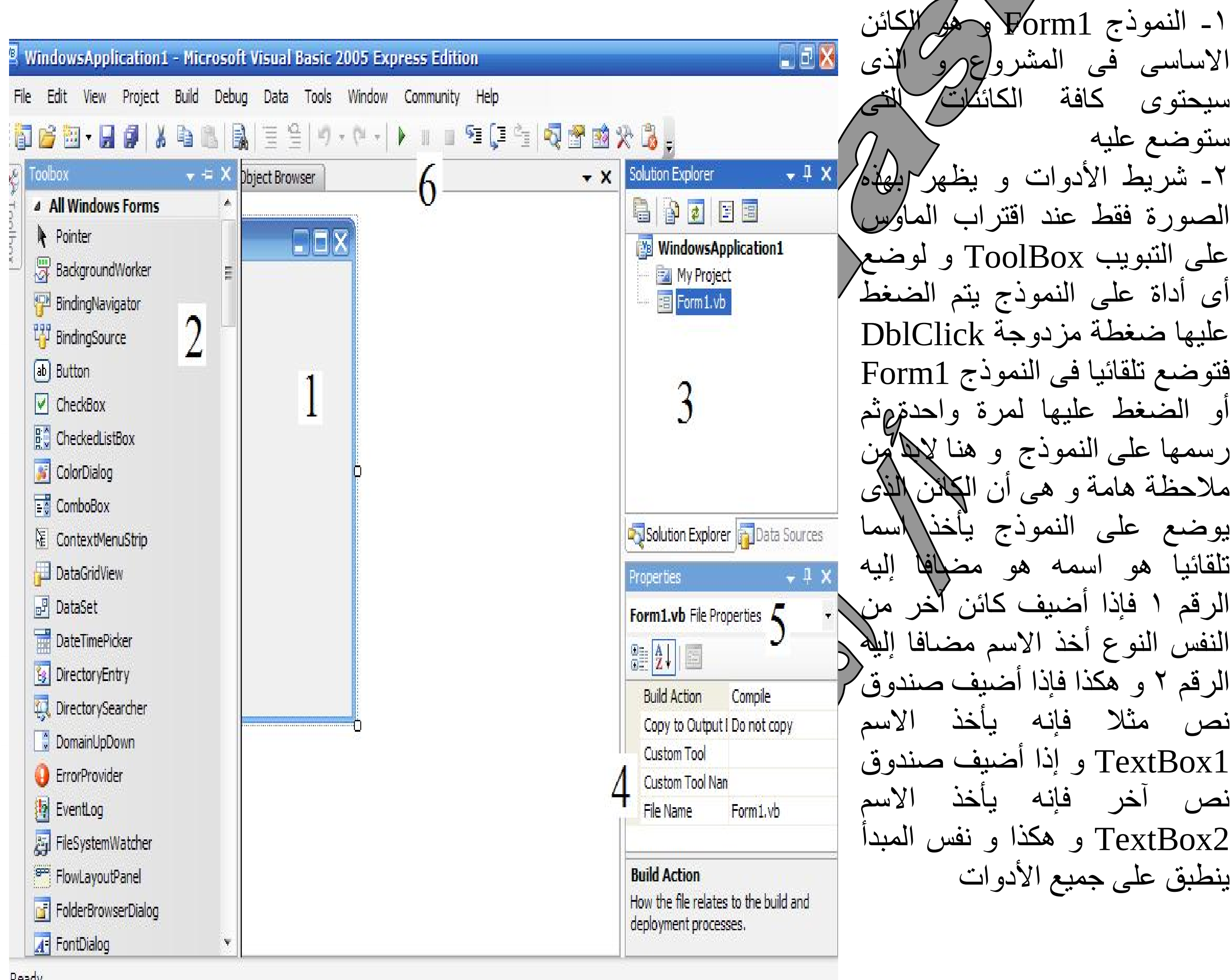
١- ***Form Designer :*** "مصمم النموذج – النافذة-" للمساعدة فى خلق المساحة اللازمة للبرنامج و هذه المساحة تسمى Form
٢- وسيلة لتسهيل عملية كتابة البرنامج و تصحيح أخطاء الكتابة و يطلق على هذه الوسيلة ***Editor***
٣- ***مترجم Compiler*** :- لتحويل البرنامج إلى لغة الآلة(لتنفيذه)
٤- ***برنامج Debugger(المصحح) :*** لتتبع و تصحيح أخطاء البرمجة
٥- ***Object Browser :*** لعرض ما تم استخدامه فى البرنامج من Classes , Objects , Events , Methods
٦- ***برنامج للمساعدة Help :*** للحصول على المساعدة فى وقت الحاجة عند الضغط على F1



مكونات الشاشة الافتتاحية قبل بدء العمل فى المشروع:-

١- المشاريع الموجودة و نريد فتحها نضغط على Project المقابلة لـ Open
٢- إنشاء مشروع جديد نضغط على Project المقابلة لـ Open
٣- قائمة آخر مشاريع تم العمل عليها فإذا كنا نريد فتح احدها نضغط على اسمه DblClick
٤- شريط الأدوات الذى يحتوى كافة الأدوات المطلوبة للمشروع
٥- نافذة الحل و سوف تُظهر هذه النافذة فيما بعد جميع عناصر المشروع الجاري العمل فيه
٦- نافذة المساعدة لعرض تعليمات بدء العمل

بعد تشغيل VB .Net تظهر الشاشة السابقة و التى تم شرح عناصرها فإذا تم اختيار الاختيار المرقم ٢ سيبدأ المعالج فى السؤال أولا عن نوع المشروع و عندها سنختار Windows Application و هو أول اختيار و محدد مسبقا و أسفل النافذة سنجد اسم المشروع و يعطى عادة الاسم WindowsApplication1 و هذا الاسم يمكن تغييره بطبيعة الحال و بعد الضغط على OK تظهر الشاشة التالية ليبدأ العمل فى المشروع



١- النموذج Form1 و هو الكائن الاساسى فى المشروع و الذى سيحتوى كافة الكائنات التى ستوضع عليه

٢- شريط الأدوات و يظهر بهذه الصورة فقط عند اقتراب الماوس على التبويب ToolBox و لوضع أى أداة على النموذج يتم الضغط عليها ضغطة مزدوجة DblClick فتوضع تلقائيا فى النموذج Form1 أو الضغط عليها لمرة واحدة ثم رسمها على النموذج و هنا لابد من ملاحظة هامة و هى أن الكائن الذى يوضع على النموذج يأخذ اسما تلقائيا هو اسمه هو مضافا إليه الرقم ١ فإذا أضيف كائن آخر من النفس النوع أخذ الاسم مضافا إليه الرقم ٢ و هكذا فإذا أضيف صندوق نص مثلا فإنه يأخذ الاسم TextBox1 و إذا أضيف صندوق نص آخر فإنه يأخذ الاسم TextBox2 و هكذا و نفس المبدأ ينطبق على جميع الأدوات

٣- نافذة الحل التى تحتوى جميع عناصر المشروع و <u>عند الضغط على العنصر فإنه يظهر أمامها إن لم يكن ظاهراً</u>

٤-نافذة الخصائص و تظهر كافة خصائص الكائن المحدد و هو فى الصورة النموذج Form1 اذا اختفت هذه النافذة تظهر بالضغط علىF4

٥- القائمة المنسدلة التى تحتوى أسماء كافة الكائنات فى المشروع ليمكن اختيار الكائن المطلوب ضبط خصائصه

٦- أزرار تشغيل و إيقاف البرنامج و لابد من التذكر دائما أن أثناء تشغيل البرنامج ينتقل البرنامج من نمط التصميم الى نمط التشغيل و فى نمط التشغيل لايمكن القيام بالعمال التى يمكن القيام بها فى نمط التصميم.

<u>هام جدا:- جميع النوافذ الظاهرة على الشاشة اذا اختفى أى منها يمكن اظهارها من خلال قائمة View</u>

الوظائف الموجودة بالقوائم :-

١- قائمة ملف File و بها كافة الأوامر الخاصة بالتعامل مع المشروع كوحدة واحدة(فتح – حفظ – اغلاق – طباعة ....الخ)

New Project	مشـــــروع جديد	Open project	فتح مشروع موجود	Save All	حفــــــــــظ البرنامج(المشروع)
Exit	الخـــروج مــن البرنامج				

٢- قائمة View و بها كافة الأوامر الخاصة بإظهار و إخفاء أشرطة الأدوات و النوافذ المختلفة و منها

Properties Window	نافذة الخصائص(٤و ٥ بالصورة)
Toolbars	أشرطة الأدوات(٦ بالصورة)
Toolbox	صندوق الأدوات(٢ بالصورة)
Solution Explorer	نافذة الحل(٣ بالصورة)

٣- بعض مفاتيح الوظائف فى برنامج فيجوال استوديو

- F4 إظهار نافذة الخصائص
- F5 تشغيل البرنامج
- F7 إظهر نافذة البرمجة (مشهد الكود)

٤- تشغيل البرنامج من خارج فيجوال استوديو يتم بعد اختيار Build من القائمة الرئيسية

أسئلة الفصل الخامس

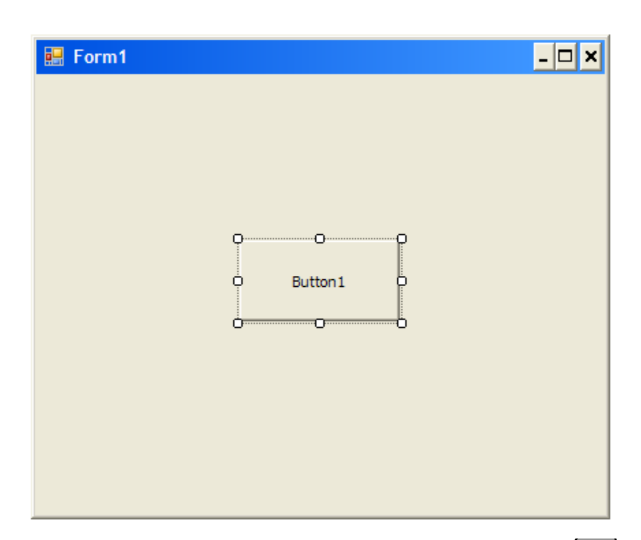
س١- ما معنى مصطلح IDE

ج١- هى الواجهة المرئية Interface للـ VB.net حيث يمكنك كتابة برامجك و عمل تطبيقات بلغة VB.net و تحتوى على الأدوات التى تمكنك من تصميم Design و تنفيذ Execute و تصحيح Debug برامجك.

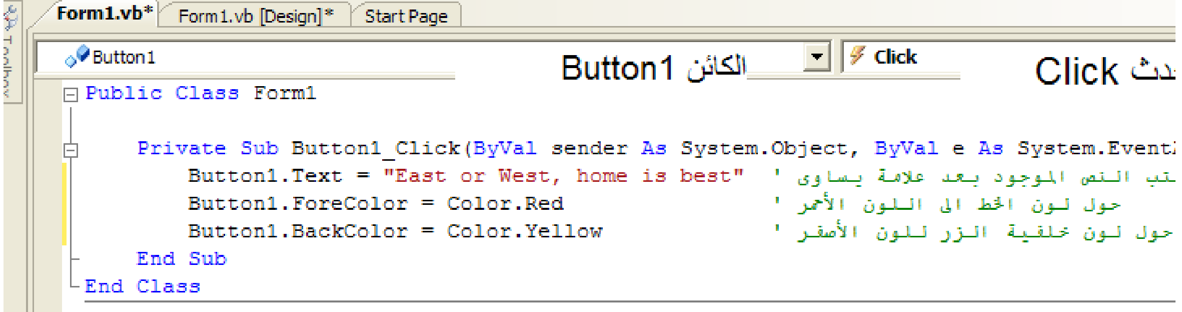
س٢- أذكر استخدامات نافذة الخصائص Properties Window و صندوق الأدوات Toolbox ؟

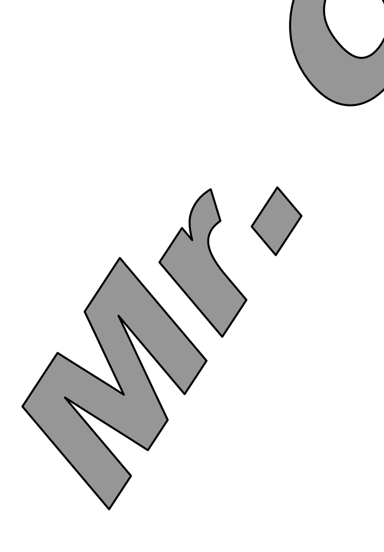
ج٢- تستخدم نافذة الخصائص فى التحكم فى خصائص الكائن Object الذى تم تنشيطه من الكائنات الموجودة فى مشاريع فيجوال بيسك. أما صندوق الأدوات Toolbox فيحتوى على مجموعة من الأدوات التى يستخدمها المبرمج فى تصميم شاشات البرنامج. و يستطيع المبرمج نسخ ما يشاء من هذه الأدوات و وضعه على النموذج (Form) الذى يقوم بتصميمه.

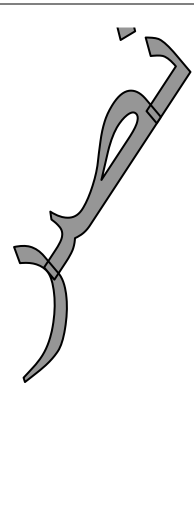
ج٣- يتم وضع الكائن Button على ال Form و سيبدو بالشكل التالى



ثم يتم الضغط ضغطا مزدوجا DblClick على الزر لتظهر نافذة البرمجة و يتم كتابة الكود الموجود كما بالشكل للحصول على المطلوب من المسألة







الفصل السادس

مقدمة إلى الأدوات

أول الأدوات و الموجودة بشكل افتراضي لا يمكن الاستغناء عنه هو أداة النموذج Form حيث يتم وضع باقي الأدوات على هذه النافذة و لذلك هي الكائن الأساسي في أي برنامج.

بعض الأدوات الشائعة الاستخدام و خواصها

أولا:- أداة الزر Button

و وظيفة هذه الأداة الرئيسية هى الاستجابة للضغط عليها من قبل المستخدم للقيام بأداء عمل محدد و عادة ما يتم تسمية هذه الأداة بالتسمية Button1, Button2 و هكذا حسب العدد و يلجأ المبرمجون إلى إعادة تسمية هذه الأداة بإضافة الثلاثة حروف btn يليها الوظيفة المخصص لها الزر مثال ذلك زر إضافة بيانات يتم تسميته btnAdd حتى يفهم من الاسم دلالة وظيفة الزر و هكذا و هذا الأمر ينطبق على باقى الأدوات.

ملاحظة:- بعض الخصائص لها نفس الوظيفة حتى لو تغير الكائن

و الجدول التالى يوضح بعض خصائص الأداة Button

Property الخاصية	Action ما تقوم به الخاصية
BackColor	معرفة أو تحديد لون الخلفية
BackgroundImage	معرفة أو تحديد الصورة التى تظهر فى خلفية الأداة
Enabled	معرفة أو تحديد قيمة معينة تبين ما إذا كانت الأداة ستستجيب لمشغل البرنامج أم لن تستجيب. و تأخذ هذه الخاصية القيمة True أو القيمة False
Font	معرفة أو تحديد نوع الخط الذى ستتم الكتاب هبه على وجه الأداة
ForeColor	معرفة أو تحديد لون الكتابة على الأداة
Image	معرفة أو تحديد الصورة التى ستظهر على الأداة
ImageAlign	معرفة أو تحديد موقع الصورة على الأداة
Name	معرفة أو تحديد اسم الأداة فى البرنامج
Text	معرفة أو تحديد ما سيكتب على وجه الأداة
TextAlign	معرفة أو تحديد موقع الكتابة على الأداة
Visible	معرفة أو تحديد قيمة معينة تبين ما إذا كانت الأداة مرئية أو غير مرئية و تأخذ هذه الخاصية القيمة True أو القيمة False

ثانيا : الأداة Label (المبين – أداة العنوان)

و تستخدم هذه الأداة لعرض كتابة لا يستطيع المستخدم أن يغير فيها و هى تستخدم عادة لعرض العناوين أو رسائل ثابتة للمستخدم .

و الجدول التالى يوضح بعض الخصائص .

Property الخاصية	**Action ما تقوم به الخاصية**
Name	معرفة أو تحديد اسم الأداة فى البرنامج و يكون مسبوقا بـ lbl
AutoSize	معرفة أو تحديد قيمة معينة تبين ما إذا كانت أبعاد الأداة ستتغير تلقائيا
BorderStyle	معرفة أو تحديد نوعية الحدود الخارجية للأداة
Font	معرفة أو تحديد نوع الخط الذى ستتم الكتاب هبه على وجه الأداة
ForeColor	معرفة أو تحديد لون الكتابة على الأداة
Image	معرفة أو تحديد الصورة التى ستظهر على الأداة
ImageAlign	معرفة أو تحديد موقع الصورة على الأداة
Text	معرفة أو تحديد ما سيكتب على وجه الأداة
TextAlign	معرفة أو تحديد موقع الكتابة على الأداة

ثالثا: الأداة TextBox (صندوق النص)

تستخدم الأداة TextBox لتعطى للمستخدم مكانا لإدخال البيانات أثناء تشغيل البرنامج.

و فيما يلى بعض الخصائص الشائعة لهذه الأداة

Property الخاصية	**Action ما تقوم به الخاصية**
Name	معرفة أو تحديد اسم الأداة فى البرنامج و يكون مسبوقا بـ txt
AutoSize	معرفة أو تحديد قيمة معينة تبين ما إذا كانت أبعاد الأداة ستتغير تلقائيا طبقا لحجم الخط المستخدم فى الكتابة.
BorderStyle	معرفة أو تحديد نوعية الحدود الخارجية للأداة
CharcterCasing	معرفة أو تحديد ما إذا كانت الأداة ستتدخل فى تعديل حالة الأحرف Capital - Upper أو Small - Lower
Font	معرفة أو تحديد نوع الخط الذى ستتم الكتاب هبه على وجه الأداة
MaxLength	معرفة أو تحديد أقصى عدد من الحروف ممكن كتابتها داخل الأداة
Modified	معرفة أو تحديد قيمة تبين ما إذا كان مستخدم البرنامج قد أجرى تعديلات على الأداة منذ بداية البرنامج
Multiline	معرفة أو تحديد قيمة تبين ما إذا كانت الأداة متعددة الأسطر
PasswordChar	معرفة أو تحديد نوع الحرف أو الشكل الذى يخفى كلمة السر Password عند إدخالها فى أداة ذات سطر واحد Single Line
ReadOnly	معرفة أو تحديد قيمة تبين ما إذا كانت محتويات الأداة للقراءة فقط، أى لا يمكن تعديلها.
ScrollBars	معرفة أو تحديد شريط الانزلاق الذى يظهر فى أداة متعددة الأسطر Multi Line
SelectedText	معرفة أو تحديد قيمة تبين الجزء الذى تم اختياره حاليا من محتويات الأداة
SelectionLength	معرفة أو تحديد قيمة عدد الأحرف التى تم اختيارها من الأداة
SelectionStart	معرفة أو تحديد نقطة الاختيار من محتويات الأداة
Text	معرفة أو تحديد ما سيكتب على وجه الأداة
TextAlign	معرفة أو تحديد موقع الكتابة على الأداة
TextLength	معرفة عدد الأحرف المكتوبة داخل الأداة
WordWrap	تبين فى حالة وجود أداة متعددة الأسطر إمكانية نقل الكلمة الزائدة عن سطر إلى بداية

سطر جديد	

بعض الوظائف و الأفعال المرتبطة بالأداة TextBox

الفعل(ما تقوم به الوظيفة) Action	**الوسيلة(الوظيفة) Method**
تضيف كتابة إلى ما هو موجود بال text box	AppendText
تمحو كل ما هو موجود بال text box	Clear
ترسل نسخة من الجزء الذى تم اختيارة من ال text box إلى الحافظة	Copy
ترسل الجزء الذى تم اختياره من text box الى الحافظة	Cut
تستبدل الجزء المختار من محتويات ال Text Box ما هو موجود فى الحافظة	Paste
تختار كل الكلام الموجود فى ال Text Box	SelectAll
الغاء آخر تعديل على الكتابة الموجودة فى ال Text Box	Undo

رابعا: الأداة MsgBox مربع الرسالة

و تستخدم لإظهار رسالة هامة للمستخدم لا بد أن يطلع عليها و يتوقف البرنامج عن العمل حتى يضغط المستخدم على زر فى مربع الرسالة.

قاعدة استخدام الأمر MsgBox

MsgBox(Arg1,[Arg2],[Arg3])

حيث Arg1 هو الرسالة التى تظهر للمستخدم

Arg2 يمثل نوع الرسالة التى ستظهر للمستخدم و يرتبط بها عدد الأزرار التى تظهر على الرسالة و كذلك نوع الأيقونة التى تظهر على الرسالة و هذا المعامل اختيارى يمكن إهماله.

Arg3 يمثل عنوان الرسالة و هو أيضا معامل اختيارى يمكن إهماله

(راجع الكتاب لمطالعة أنواع الرسائل)

الأحداث

يعرف الحدث بأنه رسالة من الكائن تعطى إشارة لحدوث حركة ما و هذا الحدث أو الحركة قد تكون ضغطة زر أو الضغط على مفتاح من لوحة المفاتيح أو كتابة نص .. الخ و يمكن أن يكون الحدث مبرمجا بشكل مسبق. و بناء على الحدث يمكن اتخاذ إجراء معين أو تجاهل هذا الحدث حسب الحاجة فى البرنامج.

حل الأسئلة

السؤال الأول

أ- هل كتابة btnCalculate كصفة Text الخاصة بزر تعتبر فكرة جيدة؟ أشرح السبب

ج أ- العبارة المذكورة يمكن استخدامها كخاصية Name للزر و ليس Text لأنها كنص يظهر على الزر لا تعنى شيئا واضحا و يمكن ببساطة كتابة Calculate فقط

ب- هل هناك أى معنى من جعل صفة Text الخاصة بالأداة Label سلسلة رمزية فارغة " empty string "؟ أشرح السبب.

جـ ب – لا معنى لجعل الخاصية Text تساوى سلسلة زمنية فارغة لأن معنى ذلك عدم ظهور شئ على الكائن Label و لكن يمكن استخدام هذه الخاصية لجعل الأداة Label خالية لبعض الوقت (أثناء بداية البرنامج مثلا) أو حدوث حدث معين و عندها يتم استخدام الكائن للكتابة عليه.

جـ - أذكر مميزات الأداة MsgBox عن الأداة Label

ج جـ - عند استخدام الأداة MsgBox فإن تنفيذ البرنامج يتوقف لحين الرد على الرسالة و هذا لا يحدث فى الأداة Label

إجابة السؤال الثانى:-

لإنشاء الواجهة الموجودة بالصورة نقوم بوضع عدد ٢ Button بالاضافة إلى الـForm الموجودة أصلا و بذلك يصبح لدينا ثلاث كائنات نقوم بتغيير خواصها كما بالجدول التالى مع اعتبار Button2 هو الزر الصغير و Button1 هو الزر الكبير.

الكائن	الخاصية	القيمة
Form1	Text	Minstry of Education
	BackColor	Coral
Button1	Text	زر صغير
	BackColor	Green
Button 2	Text	Big Button
	BackColor	Yellow

إجابة السؤال الثالث:-

يتم برمجة الزر btnYes فى الحدث Click كالتالى

```
lblReply.Text="There is no rose without a thorn"
```

و يتم برمجة الزر btnNo فى الحدث Click كالتالى:

```
lblReply.Text=" What be cured must be endured "
```


تمت
Mr. Osama Nasr
أ/ أسامة نصر